# MCA PREAMBLE DOCUMENT CURRICULUM

FOR

## MASTER OF COMPUTER APPLICATIONS (MCA)

## JAMIA MILLIA ISLAMIA

**APPROVED BY**

**BOARD OF STUDIES (26TH APRIL 2016)**

**DEPARTMENT OF COMPUTER SCIENCE**

**JAMIA MILLIA ISLAMIA**

**NEW DELHI**

# CONTENTS

# PREFACE

The present document is mainly the curriculum of Master of Computer Applications (MCA) Programme which is approved by the Board of Studies of Department of Computer Science Jamia Millia Islamia New Delhi on 26th April 2016 and authorising the Chairman, BoS to make minor modifications and corrections whenever required in future enable it to implement with effect from Academic Session 2016-17. Later, details about PGDCA programme and PhD Course work are also included as Annexure I and II.

The 3-years (6-semesters) MCA programme was first designed and proposed by Prof. V. Rajaraman Committee way back in 1982. It was a joint effort of Department of Electronics, Government of India, and University Grant Commission (UGC) where the objective of the Programme was to fulfil the massive needs of Programmer and Software Engineers to be created by Indian Industries by the end of the century 2000 due to Computerization call world over. The very first curriculum then designed was implemented with the sanctions of very selective Universities, from very next year 1983, onwards, such as JNU and AMU, besides few others. The curriculum later revised by a working group of Indian Society of Technical Education (ISTE) in 1990. Subsequently, the All India Board of Computer Science, Engg,/Tech. and Applications (AIBCSA) was set up by AICTE constituted a Sub-Committee with Prof. D. V. R. Vithal as its Chairman to suggest a further revised curriculum for the MCA programme. With fast growing changes in the subject like Computer Science and Information Technology, various BoS of the universities kept on changing the curriculum to suit the need of the Industries. Moreover, later on MCA degree also became a recognized degree for teaching in University Department and Engineering Colleges (initially it was not the eligibility criterion for Teaching) which made it a doubly edge programme –Industrial need of Software Engineers and Lecturer / Assistant Professor in Higher / Technical Education in India. The present document is also an effort of the same chain to develop and design the curriculum of MCA keeping in minds Software Professionals needed in the Indian Industries beside the need of effective teachers in University or Higher Education System.

MCA Programme has brought some of the unique features with it which was not there in the past with any other Academic degree or Programmes. The unique features of MCA curriculum was and it is maintained till date is, (though structured followed in many other programmes later) a **Professional Master Terminal Degree Programme of six-semester (three-years)** where the last semester is totally to be spent in the industries while still remaining academically connected and evaluated by the Academic Institution concerned. The idea to keep last semester in the Industry to complete "Industrial Project", is to expose the students to the industry environment while they are still in the University / Academic Institution, so their sailing in Professional and Industrial world be a smooth transition without any cultural shock.

Attempts are made to adopt UGC latest guideline on CBCS/CBSE/CBAE in incorporating the present MCA curriculum in the best possible manner keeping in mind the MCA programme objectives being Applications Oriented and Software Industries specific.

The editor welcomes any suggestions, comments and constructive criticism which may help to improve this document. The copyright of this document lies with the author and the institute – Jamia Millia Islamia, therefore reproduction of the same in part or full, without written permission may invite legal action.

(Dr Syed Afzal Murtaza Rizvi)
Chairman, BoS

# ACKNOWLEDGEMENTS

Any sincere effort to reach Quality outcome has to involve various sources with the well-directed multiple thoughts and documents of different minds who have the relevant experience, capacity and capability to contribute in the present space of time, for any act and objective, more so once it comes to Higher and professional education in the subject of Computer Science and Applications. The same was applicable while developing and re-structuring the curriculum of MCA keeping in mind the objectives of the MCA programme in 21st century.

Many people were involved in this herculean task which was based on the survey and to get direct opinions from the established subject experts in the process of development of this curriculum. Therefore the task cannot be completed or rather it will be injustice, unless it acknowledges all concerned who have contributed to it, in whatsoever manner.

First of all the exercises started in house within the Department seeking opinion of individual Faculty Members and Senior Scholars to begin with. My sincere thanks to ALL the Faculty Members and Scholars of the Department for their opinions and contributions, viz., Prof S M Khurshid Quadri, Prof Muhammad Abulaish, Dr Monica Mehrotra, Dr Mohd Nazir, Dr Rajender Kumar, Dr Syed Zeeshan Hussain, Dr Mansaf Alam, Dr Jahiruddin, Dr Suriaya Jabin, Mr Taran Singh Bharti and Dr Khalid Raza, with due mention of supporting staff Dr Israr Ahmad, Mr Abdul Aziz and Mr Hasan Mujtaba Kidwai. Specific names deserve mentioning amongst scholars in this regard are Mr Aga Syed Shafat Ali, Mr P K Yadav, Mr Chauhan, Mr Sanjeev, Mr Naem Ahmad and Mr Mudasir Ahmad Wani. Due contributions are made and worth mentioning from other Departments of our University, such as Prof S M Khursheed Haider from Mathematics, Prof Tanvir Ahmad and Mr Faiyaz Ahmad from Computer Engineering and Dr Shane Kazim Naqvi, Additional Director, FTK-CIT, JMI.

Many Senior Professors of various Universities and IT Professionals and Alumni from Industries have interacted and contributed a lot which has helped us to keep the curriculum well connected with the Industry's requirements, which is one of the major Goals of the MCA programme. We are specifically thankful to the following for their contributions and suggestions which has led to form this standard syllabus for MCA to emerge as one of the best at National Level: Prof A K Pujari, Former Professor of Hyderabad University and now Vice Chancellor, Central University of Rajasthan, Mr Shams-ul Haq, Hewlett Packard Enterprise, Prof R K Agarwal and Prof. D.K. Lobiyal from JNU, Prof Sunil Kumar Mutto, and Prof Naveen Kumar from Delhi University, Prof Praveen Chandra and Dr Sanjay Malik from GGSSIPU, Prof Rajender Singh Chhillar from M D University, Prof S. Maheshwari, Prof Suhail Mustajab, Prof Nasim Ahmad from AMU, Prof Vipin Dixit from Central University BBAU, Lucknow, Prof G. Hemantha Kumar of Mysore University, Prof A. M. Gonsai, Saurashtra University, Gujarat, Prof Afshar Alam, Prof Ranjit Biswas and Dr S Imtiyaz Husain from Jamia Hamdard University. Besides my scholars, who have already made their marks in different institutions, such as Prof Saba Hilal, Director, Prof Pankaj Agarwal, Dean, IMS, Ghaziabad, Prof Adesh Pandey, Head, IT and Prof Vineet K Sharma, Head, Computer Engineering KIET of U.P. Technical University.

I shall be failing in my duties if I do not mention the names of my family members who have provided continuous support at every stage without which the work may not have been completed with effective results. They are my wife, Prof Halima Sadia Rizvi, Professor and Former Head of Economics Department of Jamia Millia Islamia, my daughter, Er Ms Neda Fatima, B.Tech. (E&C), and my son, Dr Syed Bilal Abbas Rizvi, MBBS.

Although the attempts are made to bring the best in terms of content as well as format and presentation of this curriculum, however, it is still very much possible that it may have errors, shortcomings and omissions, for which the undersigned is entirely responsible. Suggestions, comments and constructive criticism are welcomes to improve this document further keeping in mind the objective of the MCA programme in the present time.

(Dr Syed Afzal Murtaza Rizvi)
Chairman, BoS

# MCA Curriculum

## 1. Introduction

The Department of Computer Science, a premier department of Jamia Millia Islamia (A central university), was established in the year 1999 with the objective of developing competent IT professionals and researchers of international standard and to primarily fulfil the requirements of booming IT industry. It is one of the well-established departments of Faculty of Natural Sciences with four eminent Professors, one Associate Professor, nine highly experienced Assistant Professors, and state of the art infrastructure,Laboratory and Departmental Library. The Department offers three postgraduate programmes, viz., the regular programmes MCA and PGDCA, and a self-financed evening programme M.Sc. (Bioinformatics) besides PhD in Computer Science and Bioinformatics.

The MCA is **Professional Master Terminal Degree Programme of six-semester (three-years)** where the last semester is totally to be spend in the industry while still academically connected and evaluated by the Department. It is designed to be a good blend of both the practical and theoretical aspects of computer science keeping in mind that they have to be Software Professionals needed in the Indian Industries beside they should also be effective teachers in Indian University or Higher Education System, globally.

The curriculum is based on the analysis and survey performed on courses taught at the top 20 best universities in the World and involving Senior Professors and Academics teaching MCAs, including Professionals and Leaders of Software Industries. The course structure carefully includes the practical aspects of computer science in the form of papers/courses based on C, C++, JAVA, J2EE, MATLAB, R, Hadoop, PROLOG, and Python and at the same time it includes theoretical aspects of computer science in the form of papers based on Machine Learning, Artificial Intelligence, Data Mining and Warehousing, Big data Analytics and Cloud Computing, Digital Image Processing and GPU programming, and Pattern Matching etc.

This commitment makes our MCA students marketable in the industry and also prepares them for advanced study of Computer Science (i.e. Ph.D.). The well designed course structure/curriculum makes sure that our MCA students stay up-to-date with current and future industry as well as research trends. The choice of courses for MCA programme prepares students for the ever growing and evolving software industry, such that they will be groomed as the fittest in the competition for the most satisfying and highest-paying jobs. Eventually, after MCA, they are expected to fulfil Industrial needs of Computer Professionals – Original motto of MCA – to bring out Round Human Being with desired competence to handle professional Life in Industrial environment.

## 2. Structure of the Curriculum

The structure of MCA programme consists of six-semester where each semester, barring sixth semester are having six course each of 4 credit hours, except the sixth semester which is of 20 credit projected course. The total credit therefore for MCA programme is = 140 Cr Hours (5 sem * 6 Courses * 4 credits each + 20 credit for 6$^{th}$ Semester industrial project).

At Global Level, Higher Educational System of India is branded very rigid Educational system as it does not provide the feature called “credit transfer”, amongst Institutions at National and International Levels. To achieve this, the first step should be to calculate credit of our programmes in such manner that it should be consistent at Global level, so the credit transfer is easy with International Institutions through credits transfer and structures. Keeping this vision in mind, attempts are made to adopt the credit calculations of our MCA programme also to be in line with the international standards and practices. That is the reason, why the same is being followed in the School of Computer and System Sciences (SCSS) JNU, New Delhi. Minor corrections in regulations will be required to be consistent with international practices - already committed by our honorable Vice Chancellor. It will help finally to achieve the said vision “Credit Transfer at Global Level”. UGC and National Level policy are already started working to achieve the same vision through the implementation of CBCS/CBSE/CBAE. In view of the above, BoS has finally approved the credit calculations proposed here in the present document which was finally adopted in the BoS meeting held on 26$^{th}$ April 2016.

Hence, all the Courses in MCA Programme are of 4 Credits irrespective being Theory or Laboratory oriented. Laboratory hours are decided on the basis of nature of the course and the needs of hands-on practice sessions required by the student to enhance sufficient degree of their skills which is quite essential for the professional course like MCA. Hence practice sessions are not considered as contact hours therefore no credits are assigned for them to be in line with international standards and practices. More specifically here P is not really practical contact hours rather P stands for Practice sessions allowed to enhance the degree of skill under the guidance of teacher concerned. The courses are either lab oriented or theory with LTP=314/2 (150 marks) or LTP= 310 (100 marks) respectively, except last semester's industrial project; CSPJ61: Industrial Project for 400 marks with 20 Credit Hours, which is spread for the entire $6^{th}$ semester.

Therefore, the Total Credit Hour of MCA Programme (30 courses @ 4 credits = 120; Plus 20 Credit for Industrial Project CSPJ61 in the last $6^{th}$ semester) will be = 140 Credit.
The lists of Courses / Papers under different categories are given below:

The choice based and elective based nature of MCA programme ideally allows students to follow their passion. Choice Based Credit System (CBCS) facilitates students to choose inter-disciplinary, intra-disciplinary courses, skill oriented courses according to their learning needs, interests and aptitude and specially designed core courses like Professional and Business Communications prepare students to master English language for communication, Interpersonal Skills Development and Group discussion etc. are to fulfill the MCA programme objective of being Software Industry specific and for smooth sailing in Industrial environment.

Off late, at national level, it was realized to bring flexibility in our Education system through credit transfer, not only at national level rather International Level. It was initiated, first to make them flexible within Faculty and within University itself through the introduction of CBCS/CBSE/CBAE Courses / papers recommended by UGC. To be in line, MCA Programme also offers these Courses / papers, viz., CBCS/CBSE/CBAE to students of other departments of P.G. level, along with MCA programme along our MCA students, for two such choice-based courses in each semester. There is provision for 5 Course of CBCS, one each CBCS courses in each semester, besides 3 Courses of Choice based Skill Enhancement (CBSE) in first three semesters, and 2 courses of Choice Based Ability Enhancement (CBAE) in the subsequent fourth and fifth semester. The very first and second Course in each semester are marked for these categories courses. For each core/skill/choice/elective based papers, pre-requisite papers are also mentioned respectively (using last two digits of Course Code), which are not mandatory for MCA students rather suggestive in nature to Guide the students to be careful in selecting the Courses of their choice, with page no and column referring to the detail syllabi of respective courses/ papers.

The students of MCA Programme are particularly advised rather cautioned that while choosing any CBCS/CBSE/CBAE categories course outside the Department, at the cost of dropping the respective course within the department, they have to check the prerequisites requirements of advance semesters. If they think they can cope up in advance semester, where prerequisites are mentioned (without going through prerequisite courses), with opting courses at other departments, dropping the corresponding courses offered under the CBCS/CBSE/CBAE categories, then they may go ahead on their own responsibility. However, right to allow or reject to transfer and register in these courses offered outside the Department lies with the Head of the Department concerned.

Since the Programme objective of Three-years MCA programme is unique being "**Application Oriented and Software Industries specific"**, in contrast with prevalent, normally two years Master Programmes, it will be difficult for them to get choice of courses offered outside the Department of Computer Science, **fulfilling MCA programme objective**. It is because of this, The Department of Computer Science has designed the Courses under the Categories of CBCS/CBSE CBAE for MCA students as such to fulfill the needs within the Department.

Further, since MCA is also eligible criterion for Teaching Faculty in University Departments and Engineering colleges offering MCA programmes, the Courses offered in the fifth semester are as such to retain connectivity with the Research frontiers Areas of Computer Science and Applications, such as Data mining, Machine Learning and soft computing, and at the same they are useful for those MCA who opts for Industrial career.

Industry-Institutions Seminars are suggested frequently to be held throughout the Academic year as a part of Academic activity which enables the students of MCA to appreciate the software development which is going on in industries in India. These seminars will help the students to face interviews having latest knowledge and skill prevalent in Industries with higher degree of confidence.

MCA programmes has to serve professional world and academic organizations both, bound to cover variety of subject in its curriculum to produce Industrial Professional and University Teacher, therefore Courses of MCA programmes under different categories, besides Core Computer Science Courses, viz., from Management, Mathematics and other allied subjects. The List of all such Courses being part of MCA programme is displayed below in Tabular form.

# List of Computer Science Core Courses, Management, Mathematics, and CBCS/CBSE/CBAE Courses

| Sr No | Computer Science Core Courses (CSCC) | Management, Mathematics and *CBCS/CBSE/CBAE |
|---|---|---|
| 1 | Computer Fundamentals **(Common with CBCS)** | Professional and Business Communications (Management) |
| 2 | Digital Logic and Computer Design | Principles of Management and Organizational Behaviour (Management) |
| 3 | Operating System and Shell Programming **(Common with CBCS)** | Leadership, Interpersonal and Group Dynamics (Management) |
| 4 | Data and File Structures | MFCS, Mathematica and MATLAB (Maths) |
| 5 | Microprocessor and Computer Architecture | Theory of Computation (Maths) |
| 6 | System Analysis and Design | Scientific and Statistical Techniques using R/FORTRAN (Maths) |
| 7 | Advanced Problem Solving using Java Programming **(Common with CBCS)** | 1. Computer Fundamentals (CBCS) |
| 8 | Software Engineering with Minor Project | 2. Operating System and Shell Programming (CBCS) |
| 9 | Analysis and Design of Algorithm | 3. Advance Problem Solving using Java Programming (CBCS) |
| 10 | Computer Network and System Administration | 4. Web Based Development using J2EE (CBCS) |
| 11 | Web based development using J2EE **(Common with CBCS)** | 5. Digital Image Processing and GPU Programming (CBCS) |
| 12 | Software Project Management with Minor Project | Problem Solving and Programming in C (CBSE) |
| 13 | BIG Data Analytic and Cloud Computing | Object Oriented Programming (OOP) in C++ (CBSE) |
| 14 | Digital Image Processing and GPU Programming **(Common with CBCS)** | DBMS with Oracle based Programming (CBSE) |
| 15 | Machine Learning and Soft Computing | Artificial Intelligence and Prolog Programming (CBAE) |
| 16 | Pattern Matching using Python Programming | Data Mining and Data warehousing (CBAE) |

* These courses are offered to the P.G. students of the other departments along with MCA students. The details are mentioned in the MCA Programme Structure document.

# List of Electives I , II and III

| Sr No | Course Code | Elective I & II<br>Course Title / Name<br>(Computer Science & IT) | Elective III<br>Course Title / Name<br>(Maths, Mgmt, Allied) |
|---|---|---|---|
| 1 | | Compiler Design | Algorithmic Economics |
| 2 | | Business Expert System Development | Applied Probability and Stochastic Analysis |
| 3 | | Modelling and Simulation | Formal Methods & Verification |
| 4 | | System Programming | Graphics & Geometry |
| 5 | | Human Computer Interaction | Mathematical Signal Processing |
| 6 | | Principles of Programming Languages | Advanced Optimization Techniques |
| 7 | | Natural Language Processing | Computer Ethics and Corporate Governance |
| 8 | | Voice and Speech Recognition | Applied Probability & Statistical Methods |
| 9 | | Parallel Algorithm and Scientific Applications | Introduction to Stochastic Processes and Modeling |
| 10 | | Robotics and Automation Systems | Markov Chains, Discrete Stochastic Processes and Applications |
| 11 | ESEC46.11 | Information Security | Fuzzy System and its Applications |
| 12 | | Principles of Microprocessor Systems | Information Theory, Information and Complexity |
| 13 | ESEC46.13 | Cryptography | Biomolecular Computation |
| 14 | | Quantum Information and Computation | Internet Marketing and E-Commerce |
| 15 | | Advanced Distributed System | Learning Organization and Knowledge Management |
| 16 | | Advanced Robotics: Navigation and Vision | Entrepreneurship: Theory and Practice -Make In India Mission |
| 17 | ESEC46.17 | Advanced Data Base Management System | Creativity and Innovation for Entrepreneur Excellence |
| 18 | | Molecular Programming | Business Process Re-Engineering |
| 19 | | Networks: Structure Economics / Social Media | Enterprise Leadership and Organizational Creativity |
| 20 | | Theorem Proving | IT ACT and Cyber Security / Law |
| 21 | | Perl / LISP/C# Programming | Computational Biology |
| 22 | | Research Methodology in Scientific Research | Fuzzy System and Entropy Optimizations |

Note: The Detail syllabus of Elective will be designed whenever offered with the approval of BoS.

**********

**Approved by ACC, DC and BoS (26th April 2016) of DCS, JMI**

**Master of Computer Applications**

**Programme Structure and Detail Syllabi**

**Semester I**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSCC11<br>CBCS11 | Computer Fundamentals | NIL | 3 1 0 | 4 | 100 | 14-18 |
| 2 | CSCC12<br>CBSE12 | Problem Solving and Programming in C | NIL | 3 1 4 | 4 | 150 | 19-24 |
| 3 | CSCC13 | Digital Logic & Computer Design | NIL | 3 1 0 | 4 | 100 | 25-30 |
| 4 | CSCC14 | MFCS, Mathematica & Matlab | NIL | 3 1 4 | 4 | 150 | 31-35 |
| 5 | CSCC15 | Professional and Business Communications | NIL | 3 1 0 | 4 | 100 | 36-40 |
| 6 | CSCC16 | Principles of Management and Organizational Behaviour | NIL | 3 1 0 | 4 | 100 | 41-45 |
| | | | | | | **700** | |

**Semester II**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSCC21<br>CBCS21 | Operating System and Shell Programming | (11,12) | 3 1 4 | 4 | 150 | 46-50 |
| 2 | CSCC22<br>CBSE22 | Object Oriented Programming in C++ | (12) | 3 1 4 | 4 | 150 | 51-55 |
| 3 | CSCC23 | Data & File Structures | (12) | 3 1 4 | 4 | 150 | 56-60 |
| 4 | CSCC24 | Microprocessor & Computer Architecture | (13) | 3 1 4 | 4 | 150 | 61-65 |
| 5 | CSCC25 | System Analysis and Design | (11) | 3 1 0 | 4 | 100 | 66-70 |
| 6 | CSCC26 | Theory of Computations | (14) | 3 1 0 | 4 | 100 | 71-75 |
| | | | | | | **800** | |

**Semester III**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSCC31<br>CBCS31 | Advance Problem Solving using Java Programming | (22) | 3 1 4 | 4 | 150 | 76-80 |
| 2 | CSCC32<br>CBSE32 | DBMS with Oracle Based Programming | (11) | 3 1 4 | 4 | 150 | 81-85 |
| 3 | CSCC33 | Software Engineering with Minor Project (Definition & Design Phase) | (23,25) | 3 1 0 | 4 | 100 | 86-90 |
| 4 | CSCC34 | Analysis and Design of Algorithm | (22,26) | 3 1 4 | 4 | 150 | 91-95 |
| 5 | CSCC35 | Computer Network and System Administrations | (21) | 3 1 4 | 4 | 150 | 96-100 |
| 6 | CSCC36 | Scientific and Statistical Techniques using Fortran / R | (14) | 3 1 4 | 4 | 150 | 101-105 |
| | | | | | | **850** | |

**Semester IV**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSCC41<br>CBCS41 | Web Based Development using J2EE | (22,31) | 3 1 4 | 4 | 150 | 106-110 |
| 2 | CSCC42<br>CSAE42 | Artificial Intelligence and Prolog Programming | (12,14) | 3 1 4 | 4 | 150 | 111-115 |
| 3 | CSCC43 | Software Project Management with Minor Project (Testing & Implementation) | (22,33) | 3 1 4 | 4 | 150 | 116-120 |
| 4 | CSCC44 | BIG DATA Analytic and Cloud Computing | (32,33) | 3 1 4 | 4 | 150 | 121-126 |
| 5 | CSCC45 | Leadership, Interpersonal and Group Dynamics | (16) | 3 1 0 | 4 | 100 | 127-131 |
| 6 | CSEC46 | Elective I (CS & IT) | | 3 1 0 | 4 | 100 | 132 |
| | | | | | | **800** | |

**Semester V**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSCC51<br>CBCS51 | Digital Image Processing and GPU Programming | (14) | 3 1 2 | 4 | 150 | 133-137 |
| 2 | CSCC52<br>CSAE52 | Data Mining and Data Warehousing | (32,42) | 3 1 2 | 4 | 150 | 138-142 |
| 3 | CSCC53 | Machine Learning and Soft Computing | (34,42) | 3 1 2 | 4 | 150 | 143-147 |
| 4 | CSCC54 | Pattern Matching using Python Programming | (12) | 3 1 2 | 4 | 150 | 148-152 |
| 5 | CSEC55 | Elective II (CS & IT) | | 3 1 0 | 4 | 100 | 153 |
| 6 | CSEC56 | Elective III (Allied) | | 3 1 0 | 4 | 100 | 153 |
| | | | | | | **800** | |

**Semester VI**

| Sr No | Course Code | Course Title / Name | Pre-requisite | L T P | Credit | Marks | Page No |
|---|---|---|---|---|---|---|---|
| 1 | CSPJ61 | Industrial Project<br>(Capstone Course) | Atleast 12 CH@ each Sem | N.A. | 20 | 400 | |

**Total Credit of MCA = 140**

**Note**: The above Structure will be effective totally for the MCA batch to be admitted in 2016-17 onwards and from 3rd Semester onwards for MCA batch admitted 2015-16 (relaxing Prerequisites conditions, if needed), both being adopted incorporated CBCS scheme, since only one year Complete Syllabi was designed for the MCA batch 2015-16.

Please be informed that first TWO papers in each semester are having TWO codes, viz. CSCC11/CBCS11. The first code is for MCA students for them it is a Core Course and second code is for the outside students, to provide them the opportunities to get themselves exposed to some Courses / Papers to our prestigious MCA programme along our MCA students. It is not an opportunity for our MCA students to register any course / paper outside the department at the cost of their core course as MCA is having Unique Programme objectives where thrust is on "Application Perspective and towards Software Specific Industries."

For further details one can refer to MCA Preamble documents which have already illustrated it more clearly.

# Course Plan and Detail Syllabi for

# Computer Fundamentals

| | | |
|---|---|---|
| Course Code | : | **CSCC11 / CBCS11** |
| Credit | : | 4 |
| L-T-P | : | 3-1-0 |
| Course Prerequisite | : | NIL |
| Course Status | : | Core course / Choice Based CS |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course is all about Computer which is an advanced electronic device with no IQ that takes raw data as input from the user and processes it under the control of set of instructions (called program), gives the result (output), and saves it for the future use. This Computer Fundamentals course covers a foundational understanding of computer hardware, software, operating systems, peripherals, programming languages, computer networks, and data representation methods etc. This course provides a general introduction to computer systems. A computer system is made up of both hardware and software. Software is another term for computer program. Software controls the computer and makes it do useful work. Hardware refers to the physical components that make up a computer system. These include the computer's processor, memory, monitor, keyboard, mouse, disk drive, printer and so on. In this course we take a brief look at the functions of the different hardware components. In addition it describes some of the essential softwares required for the operation of a computer system.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. To give a general understanding of how a computer works
2. To introduce different programming paradigms
3. To prepare students for future courses
4. To describe the fetch execute cycle of a computer

5. To understand the different types of data representation methods which may be stored within a computer memory

## UNITWISE SYLLABUS

1. **Computing Concepts:** Basic Computing Systems, Layers of a Computing System, History of Computing, History of Computing Software, Stored-Program Concept and von Neumann Architecture. Fetch-Execute Cycle, Input-Output Devices, Touch Screens, Mouse, Keyboard etc. Data Representation and Number Systems: Binary Values and Computers, Data and Computers, Analog and Digital Data; Binary Representation. Number Systems: Binary, Octal, Decimal, and Hexadecimal. Conversions of Data from one Number System to another Number System. Representation of Numeric Data– Negatives and Real Data Representation. Representing Texts-ASCII and Unicode Character Sets. Binary Arithmetic–Addition and Subtraction of Numbers in Different Number Systems.
2. **Gates and Circuits:** Computers and Electricity; Logic Gates – AND, OR, NOT, XOR, NAND and NOR Gates. Gate Processing; Gates with More Inputs; Constructing Gates; Transistors; Circuits–Combinatorial Circuits: Adders and Multiplexers. Circuit as Memory; Integrated Circuits; CPU Chips. Memory Hierarchy: Registers, Cache, ROM, RAM, ROM BIOS/Firmware, Secondary, Tertiary Storage Devices, and their Relative Characteristics
3. **Programming Languages:** Computer Operations; Levels of Abstraction; Machine Language; Virtual Computers; Assembly Language; Pseudo-Operations; Low Level Programming Example; Programming Language Paradigms; Functionality of Imperative Languages; Boolean Expressions; Strong Typing; Input-Output Structures; Control Structures; Composite Data Types and Object Oriented Concepts, System Programs: Compilers; Interpreters; Loader, Linker, and Operating Systems.
4. **Operating Systems:** Introduction and Examples of Operating Systems, Roles of Operating Systems: Memory, Process, and CPU Management. Batch Processing, Categories of Operating Systems: Timesharing, Multitasking, Multithreading, and Multiprogramming Operating Systems. Process States and Process Control Block.
5. **Computer Networks:** Introduction to Computer Networks. Network Topologies, Types of Networks, Internet vs. Intranet, Switch, Bridge, and Routers. Firewalls, Network Protocols, TCP/IP, Network, Addresses, Domain Name System, Internet services: E-mail, FTP, etc. The World Wide Web: Introduction to World Wide Web. Search Engines, Instant Messaging, Weblogs, Cookies, Introduction to Markup Languages, HTML and XML. Webpages and their Components, Categories of Webpages. Java Applets, Java Server Pages.

### Text Books


1. Dale & Lewis: Computer Science Illuminated, 3rd ed, Narosa Publishing House, 2007
2. M. Morris Mano: Computer System Architecture, 3rd Pearson Education Publication.

3. Kedall & Kendall: Systems Analysis and Design, 5th ed, Prentice Hall India, 2005
4. Ankita Goel, Computer Fundamentals, Pearson Publication.

**Reference Books**

1. Rajaraman: Fundamentals of Computers, 4th ed, Prentice Hall India, 2007
2. ITL Esl: Introduction to Computer Science, 1st, Pearson Education, 2009
3. R. G. Dromey: How to Solve it by Computer, 2nd Ed., Pearson Education
4. E. Balaguruswamy: Computing Fundamentals and C Programming,Mc Graw Hill Publication.

**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Basic Computing Systems, Layers of a Computing System, History of Computing, History of Computing Software, Stored-Program Concept and von Neumann Architecture, Fetch-Execute Cycle |
| **Week 2:** | Input-Output Devices, Touch Screens, Mouse, Keyboard etc. |
| **Week 3:** | Data Representation and Number Systems: Binary Values and Computers, Data and Computers, Analog and Digital Data; Binary Representation |
| **Week 4:** | Number Systems: Binary, Octal, Decimal, and Hexadecimal. Conversions of Data from one Number System to another Number System. |
| **Week 5:** | Representation of Numeric Data– Negatives and Real Data Representation. Representing Texts-ASCII and Unicode Character Sets. Binary Arithmetic–Addition and Subtraction of Numbers in Different Number Systems. |
| **Week 6:** | Computers and Electricity; Logic Gates – AND, OR, NOT, XOR, NAND and NOR Gates. Gate Processing; |
| **Week 7:** | Gates with More Inputs; Constructing Gates; Transistors; Circuits–Combinatorial Circuits: Adders and Multiplexers. |
| **Week 8:** | Circuit as Memory; Integrated Circuits; CPU Chips. Memory Hierarchy: Registers, Cache, ROM, RAM, ROM BIOS/Firmware, Secondary, Tertiary Storage Devices, and their Relative Characteristics |
| **Week 9:** | Computer Operations; Levels of Abstraction; Machine Language; Virtual Computers; Assembly Language; Pseudo-Operations; Low Level Programming Example; |

Programming Language Paradigms; Functionality of Imperative Languages;

**Week 10:** Boolean Expressions; Strong Typing; Input-Output Structures; Control Structures; Composite Data Types and Object Oriented Concepts, System Programs: Compilers; Interpreters; Loader, Linker, and Operating Systems.

**Week 11:** Introduction and Examples of Operating Systems, Roles of Operating Systems: Memory, Process, and CPU Management. Batch Processing,

**Week 12:** Categories of Operating Systems: Timesharing, Multitasking, Multithreading, and Multiprogramming Operating Systems. Process States and Process Control Block, Introduction to Computer Networks. Network Topologies, Types of Networks, Internet vs. Intranet, Switch, Bridge, and Routers. Firewalls, Network Protocols, TCP/IP, Network, Addresses, Domain Name System, Internet services: E-mail, FTP, etc.

**Week 13:** The World Wide Web: Introduction to World Wide Web. Search Engines, Instant Messaging, Weblogs, Cookies, Introduction to Markup Languages, HTML and XML. Webpages and their Components, Categories of Webpages. Java Applets, Java Server Pages.

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Problem Solving and Programming in C

| | | |
|---|---|---|
| Course Number | : | **CSCC12 / CBSE12** |
| LTP | | 3-1-4 |
| Credit Hours | : | 4 |
| Course Prerequisite | : | NIL |
| Course Status | : | Core course / Choice Based Skill Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course teaches C programming by solving a variety of standard problems. Learning programming is never a theoretical exercise; it is augmented by some basic as well as advanced programming problems. It is as similar as saying a person cannot learn swimming by just reading a book on how to swim. The more often one will jump into water, the better swimmer one will become. The same is believed to be to true for programming, the more problems students will be exposed to try to tackle, the better programmer they will become. This course introduces students to the programming of computing systems. The main objective of this course is to give them exposure to basic concepts in programming using a high-level language, which in the case of this course is the C programming language. The main motive is not to enable students writing C program but to enhance their logical thinking and reasoning power so that they can attack any new problem with a very positive attitude.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. To introduce C as a foundation for further study of programming languages such C++, C#, Python, and Java in particular.
2. To use Array, Structure and Union data structures to represent lists and table of values.

3. To be able to use pointers to fetch and process data at run-time and to dynamically allocate memory at run-time.
4. To implement one player and two player games such as Tic-Tac-Toe, NIM, and Sudoku etc.
5. To be able to design applications involving database stored at back-end in the form of text file.

## UNITWISE SYLLABUS

1. **Problem Solving Approach and Basics of C:** Introduction to Programs and Algorithms; Problem Solving Aspect (Algorithm Devising); Algorithm Design Aspect (Top-down Design); Algorithm Implementation; Program Verification. Fundamental Algorithms – Exchanging the Values of Two Variables, Counting, Summation of a Set of Numbers, Factorial Computation, Sine Function Computation, Generation of the Fibonacci Sequence, Reversing the Digits of an Integer, Base Conversion, etc. Flowchart. Basics of C: Character Set; Keywords; Identifier, Constants, and Variables; Constant Types – Numeric and Character Constants; Data Types and Range of Values – Character, Integer and Floating Point; Signed, Unsigned, Short, and Long Integers; Data Declaration and Definition, Operator & Expression–Arithmetic, Relational, Logical, Increment, Decrement, Assignment, Conditional, and Bitwise Operators; Precedence & Associability of Operators
2. **I/O & Control Structures:** Managing Console I/O–Reading and Writing Characters, Integers, Floating Point Numbers and Strings; Formatted I/O**,** Decision Making (Branching) Structures–If Statement, If-Else Statement, Nested If-Else Statement, Else-If Ladder, Switch Statement, Goto Statement; Looping Structures – While Statement, Do-While Statement, For Statement, Continue and Break Statements. Functions: Library Functions;
3. **User-Defined Functions & Arrays:** Function Declaration (Prototype) and Function Definition; Function Arguments – Dummy, Actual and Formal Arguments; Local and Global Variables; Function Calls – Call by Value and Call by Reference; Returning Multiple Values from a Function, Recursion and Recursive Functions, Storage Class & Scope of Variables – Automatic Storage, Extern Storage, Static Storage, Register Storage, Single Dimensional Arrays; Accessing Array Elements; Initializing an Array; Multidimensional Arrays; Initializing Multidimensional Arrays; Memory Representation; Accessing Multidimensional Array Elements;
4. **Strings & Pointers:** Array of Characters; String Manipulation Functions; Introduction to Pointers; Pointer Variable Declarations and Initializations; Null Pointer; Constant Pointers; Void Pointer; Pointer Operators; Pointer Arithmetic; Application of Pointers; Dynamic Memory Allocations: malloc, calloc, realloc and free functions; Implementation of One Dimensional Array Using Pointers; Implementation of Two Dimensional Array Using Pointers: Array of Pointers and Pointers to Arrays Representations; Pointers and Strings; De-Referencing Pointers; Pointer to Pointer; Pointer to Functions.

5. **Structure, Union, Enumeration and Files:** Structure Declaration and Initialization; Accessing Structure Members, Structure Assignments; Array of Structures and Arrays within Structures, Nested Structures; Structure as Function Arguments; Structure Pointer; Unions; Difference between Structure and Union; Bit-Fields; Introduction to File; Text and Binary Files; Defining, Opening and Closing Files; I/O Operations on Files, Error Handling During I/O Operations, Random Access to Files, Command Line Arguments.

**Text Books**


1. E. Balagrusamy: Programming in ANSI C, 7th Ed., Tata McGraw Hill, 2013
2. Gottfried, Programming in C – Schaum Series, 3rd edition, TMH publication, 2014
3. Brian W. Kernighan, Dennis M. Ritchie: The C Programming Language, 2nd Edition, Prentice Hall, 1988.


**Reference Books**


1. R. G. Dromey: How to Solve it by Computer, 2nd Ed., Pearson Education
2. Mike Banahan, Declan Brady and Doran: The C Book, 2nd edition, Addison Wesley, 1991.
3. Deitel & Deitel: C – How to Program, 6th Ed., Pearson Education
4. Forouzan and Gilberg, Computer Science: A Structured Programming Approach using C, Publisher: Course Technology; 3rd edition, 2006.


## Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction to Programs and Algorithms; Problem Solving Aspect (Algorithm Devising); Algorithm Design Aspect (Top-down Design); Algorithm Implementation; Program Verification |
| **Week 2:** | Flow charting and different example problems |
| **Week 3:** | Basics of C: Character Set; Keywords; Identifier, Constants, and Variables; Constant Types – Numeric and Character Constants; Data Types and Range of Values |
| **Week 4:** | Operator & Expression–Arithmetic, Relational, Logical, Increment, Decrement, Assignment, Conditional, and Bitwise Operators; Precedence & Associability of |

Operators

**Week 5:** Managing Console I/O–Reading and Writing Characters, Integers, Floating Point Numbers and Strings; Formatted I/O**,** Decision Making (Branching) Structures

**Week 6:** Goto Statement; Looping Structures – While Statement, Do-While Statement, For Statement, Continue and Break Statements

**Week 7:** Library functions and user-defined functions

**Week 8:** Recursion and Recursive Functions

**Week 9:** Storage Class & Scope of Variables, Single and Multi-Dimensional Arrays

**Week 10:** Introduction to Pointers; Pointer Variable Declarations and Initializations; Null Pointer; Constant Pointers; Void Pointer; Pointer Operators; Pointer Arithmetic;

**Week 11:** Array of Characters; String Manipulation Functions

**Week 12:** Array of Pointers and Pointers to Arrays Representations; Pointers and Strings; De-Referencing Pointers; Pointer to Pointer; Pointer to Functions

**Week 13:** Structure, Union and applications, File handling in C and applications

**Week 14:** Revision

**c) LAB: Implementation of at least one specific assignment concerning each of the following:**

- DOS and Windows: File handling, directory structures, file permissions, Intro to turbo C IDE and using Debugger.
- Creating and editing simple C program, Compilation and execution with variables and expressions.
- Implementation of Precedence of operators, Type casting, Decision control structures, Loop controls and Case control structure
- Unconditional jumps— break, continue, goto.
- Random number generation and simulating throw of a dice, game simulations-NIM, Craps, Cards etc
- Different mathematical operations using <math.h>, string functions using <string.h> etc .
- Modular program development using functions., Arrays and matrix operations and Recursion
- Pointers, address operators and pointer arithmetic; Pointers, arrays, and functions, Double referential pointers.
- Structures and Unions, Accessing their members; Dynamic memory allocation/deallocation, Self-Referential Structures, Dynamic memory allocation and deallocations and Bit fields

- File handling, File operations, Standard streams & Command line arguments, Enumerations and Preprocessors.

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for
# Digital Logic and Computer Design

| | | |
|---|---|---|
| Course Code | : | **CSCC13** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | Nil |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course will cover combinational and sequential circuit design, in detail. The implementation of different gates will also be covered. In addition, the basic concepts of digital logic design will be an essential part of the curriculum. New concepts like threshold circuit design and T-Gate implementation for all gates will also be introduced.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Design sequential circuit using different FF.
2. Develop understanding of combinational circuit designing.
3. Learn to design the different types of counters using different types of FF.
4. Understand different types of registers.

## UNITWISE SYLLABUS

**1 Information Representation:** Number Systems - Binary, Octal, Decimal, and Hexa-Decimal; Number Base Conversions; Binary Arithmetic; Complements: (r-1)'s Complement, r's Complement, Subtraction using Complements; Floating Number-Fixed-point Representation, Floating-point Representation; Binary Codes for Decimal Digits: BCD Code, Excess-3 Code, 84-2-1 Code, 2421 Code, Reflected Code; Error Detection

Code; Character Representation – ASCII, EBCDIC.

2 **Boolean Algebra, and Logic Gates:** Boolean Algebra-Basic Definitions, Huntington's Postulate, Switching Algebra, Basic Theorems and Properties of Boolean Algebra; Boolean Functions: Basic Definition, Literals, Minimization of Boolean Functions by Algebraic Manipulation, Complement of a Boolean Function; Canonical and Standard Forms: Minterms and Maxterms, Boolean Function as a Sum of the Minterms, Boolean Function as a Product of Maxterms, Conversion Between Canonical Forms, Standard Form of a Boolean Function; Other Logical Operations; Digital Logic Gates: Basic Gates – AND, OR, NOT; Universal Gates – NAND, NOR; Other Gates – XOR, XNOR, AND-OR-INVERT, and OR-AND-INVERT; Implementation of Boolean Functions.

3 **Simplification of Boolean Functions:** Karnaugh Maps (K-Map) Method: Two Variable K-Map, Three Variable K-Map, Four Variable K-Map, Five Variable K-Map, Product of Sum Simplification, Don't Care Condition, Simplification of a Boolean Function with Don't Care; Tabulation Method: Determination of Prime-Implicants, Selection of Essential Prime-Implicants.

4 **Combinational Logic:** Overview of Combinational Logic; Combinational Logic Design Procedure; Design of Some Standard Combinational Circuits: Half Adder, Full Adder, Half Subtractor, Full Subtractor, Code Conversion; Combinational Logic with MSI and LSI: Binary Parallel Adder, Decimal Adder, BCD Adder, Magnitude Comparator, Decoders, Encoder, Multiplexers, De-multiplexer, Read Only Memory (ROM), Programmable Logic Array (PLA).

5 **Sequential Logic:** Flip-Flops: RS Flip Flop, Clocked RS, JK Flip Flop, Master Slave JK Flip Flop, D Type Flip Flop, T Type Flip Flop; Analysis of Clocked Sequential Circuits: State Table, State Diagram, State Equations, Flip Flop Input Functions; Flip Flop Characteristic Tables; Flip Flop Excitation Tables; Design of Sequential Circuits.

**Registers, Counters and Threshold Circuit:** Registers: Register with Parallel Load, Shift Registers, Bidirectional Shift Register with Parallel Load, Serial Addition using Shift Registers; Counters: Ripple Counters, Binary Ripple Counters, BCD Ripple Counters, Synchronous Counters, Binary Synchronous Counter, Binary Synchronous Up-Down Counter, Binary Counter with Parallel Load, Johnson Counter. Threshold Logic Circuit, Threshold Gate (T-gate), Input output relation Table, Implementation of Conventional gates with T gate, AND gate Implementation with T gate, OR gate Implementation with T gate, NOT gate Implementation with T gate, NAND gate Implementation with T gate, NOR gate Implementation with T gate.

**Text Books**

1 M.A & B.A: Digital Logic Design, Prentice Hall of India(PHI), 2016

2 M. Morris Mano: Digital Logic and Computer Design, Prentice Hall of India

3. Roth: Fundamentals of logic design, 5th Edition,Thomson.

4. M. Rafiquzzaman John Wile: Fundamentals of Digital Logic & Micro Computer Design, 5TH Edition,

**Reference Books**

1. V. Rajaraman & T. Radhakrishnan: An Introduction to Digital Computer Design, PHI.
2. Donald e Givone: Digital Principles and Design, Tata McGraw Hill
3. Zvi. Kohavi: Switching and Finite Automata Theory , Tata McGraw Hill.
4. C.V.S. Rao: Switching and Logic Design, Pearson Education.

**c) Outline**

**Week** **Topics**

**Week 1:** Number Systems - Binary, Octal, Decimal, and Hexa-Decimal; Number Base Conversions; Binary Arithmetic; Complements: (r-1)'s Complement, r's Complement, Subtraction using Complements;

**Week 2:** Floating Number-Fixed-point Representation, Floating-point Representation; Binary Codes for Decimal Digits: BCD Code

**Week 3:** Excess-3 Code, 84-2-1 Code, 2421 Code, Reflected Code; Error Detection Code; Character Representation – ASCII, EBCDIC

**Week 4:** Boolean Algebra-Basic Definitions, Huntington's Postulate, Switching Algebra, Basic Theorems and Properties of Boolean Algebra; Boolean Functions: Basic Definition, Literals, Minimization of Boolean Functions by Algebraic Manipulation, Complement of a Boolean Function

**Week 5:** Canonical and Standard Forms: Minterms and Maxterms, Boolean Function as a Sum of the Minterms, Boolean Function as a Product of Maxterms, Conversion Between Canonical Forms, Standard Form of a Boolean Function

**Week 6:** Other Logical Operations; Digital Logic Gates: Basic Gates – AND, OR, NOT; Universal Gates – NAND, NOR; Other Gates – XOR, XNOR, AND-OR-INVERT, and OR-AND-INVERT; Implementation of Boolean Functions.

**Week 7:** Karnaugh Maps (K-Map) Method: Two Variable K-Map, Three Variable K-Map, Four Variable K-Map, Five Variable K-Map,

**Week 8:** Product of Sum Simplification, Don't Care Condition, Simplification of a Boolean Function with Don't Care; Tabulation Method: Determination of Prime-Implicants, Selection of Essential Prime-Implicants.

**Week 9:** **:** Overview of Combinational Logic; Combinational Logic Design Procedure; Design of Some Standard Combinational Circuits: Half Adder, Full Adder, Half Subtractor, Full Subtractor, Code Conversion;

**Week 10:** Combinational Logic with MSI and LSI: Binary Parallel Adder, Decimal Adder, BCD Adder, Magnitude Comparator, Decoders, Encoder, Multiplexers, De-multiplexer, Read Only Memory (ROM), Programmable Logic Array (PLA).

**Week 11:** Flip-Flops: RS Flip Flop, Clocked RS, JK Flip Flop, Master Slave JK Flip Flop, D Type Flip Flop, T Type Flip Flop; Analysis of Clocked Sequential Circuits: State Table, State Diagram, State Equations, Flip Flop Input Functions; Flip Flop Characteristic Tables; Flip Flop Excitation Tables; Design of Sequential Circuits.

**Week 12:** Registers: Register with Parallel Load, Shift Registers, Bidirectional Shift Register with Parallel Load, Serial Addition using Shift Registers; Counters: Ripple Counters, Binary Ripple Counters, BCD Ripple Counters, Synchronous Counters, Binary Synchronous Counter, Binary Synchronous Up-Down Counter, Binary Counter with Parallel Load, Johnson Counter.

**Week 13:** Threshold Logic Circuit, Threshold Gate (T-gate), Input output relation Table, Implementation of Conventional gates with T gate, AND gate Implementation with T gate, OR gate Implementation with T gate, NOT gate Implementation with T gate, NAND gate Implementation with T gate, NOR gate Implementation with T gate.

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Mathematical Foundations of Computer Science (MFCS), Mathematica and Matlab

| | | |
|---|---|---|
| Course Code | : | **CSCC14** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | Nil |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** The objective of this course is to learn a particular set of mathematical facts and how to apply them in real life examples. It teaches students how to think logically and mathematically. To achieve these goals the mathematical reasoning and ways to solve problems are required. Topics include Sets, relations to lattice, reasoning, combinatory and graph.

**b) Objectives:** Upon completion of this course, students will be able to do the following:

1. Solve the problem based on set and relations.
2. Partitioning of set based on equivalent relations.
3. Understand some basic algebraic structures.
4. Solve the reasoning problems.
5. Understand the graphs.

## UNITWISE SYLLABUS

1. **Introduction:** Set, Multi-set and Sequences; Type of sets, Set Operations, Power Set, Cartesian Products, Relation, Representation of relation, composition of relations, Functions, Types of Functions, Inverse of a functions, Compositions of functions, function representation, Sequences, Special Integer Sequences, Summations.
2. **Special Relations:** Equivalence Relation, Reflexive, Symmetric and Transitive Closure, Transitive Closure and Warshall's Algorithm; Equivalence Classes and Partitions; Partial Ordering, Lexicographic Order, Hasse Diagram, Maximal and Minimal Elements, Lattices.
3. **Logic, Reasoning and Inferences:** Foundations of Logic; Propositions, Conditional Statements, Bi-conditionals, Truth Table, Precedence of Logical Operators, Translating English Sentences, System Specifications, Logic Puzzles, Propositional, Logical Equivalences, De Morgan's Laws, and Construction of New Logical Equivalences; Predicates and Quantifiers – Predicates, Quantifiers, Universal Quantifiers, Quantifiers with Restricted Domains, Precedence, Binding Variables, Logical Equivalences, Negating Quantified Expressions, Translation to English Expressions, Rules of Inferences – Valid Arguments, Rules of Inferences for Propositional Logic, Building Arguments.
4. **Counting:** Basic Counting Principles, Simple Counting Problems, Complex Counting Problems and Examples; The Inclusion-Exclusion Principle, Tree Diagrams; The Pigeonhole Principle; The Generalized PHP; Applications of Pigeonhole Principle; Permutations, r-Permutations, Combinations, r-Combinations, Binomial Coefficients, Examples and Applications; Binomial Coefficients, Binomial Theorem, Expression, and other Identities; Permutations and Combinations with Repetition, Permutations with Indistinguishable Objects, Distributing Events into Boxes; Generating Permutations and Combinations.
5. **Graphs:** Graphs, Graph Models and Terminologies; Types of Graphs; Graph Isomorphism; Euler and Hamiltonian Paths and Circuits; Spanning tree, Number of spanning tree of a graph, Some graph algorithms.

### Text Books


1. Tremblay and Manohar: Discrete Mathematical Structures, Tata McGraw Hill.
2. Rosen: Discrete Mathematics and its Applications with Combinatorics and Graph Theory, TMH.
3. Amos Gilat: Matlab, An Introduction With Applications, Wiley India Publication, 2003.


### Reference Books


1. Kolman, et al: Discrete Mathematical Structures, Pearson Education
2. Johnsonbaugh: Discrete Mathematics, Pearson India
3. Kevin Ferland: Discrete Mathematical Structures, Cengage Learning

## c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Set, Multi-set and Sequences; Type of sets, Set Operations, Power Set, Cartesian Products |
| **Week 2:** | Relation, Representation of relation, composition of relations, Functions, Types of Functions, Inverse of a functions |
| **Week 3:** | Compositions of functions, function representation, Sequences, Special Integer Sequences, Summations |
| **Week 4:** | Equivalence Relation, Reflexive, Symmetric and Transitive Closure, Transitive Closure and Warshall's Algorithm |
| **Week 5:** | Equivalence Classes and Partitions; Partial Ordering, Lexicographic Order, Hasse Diagram |
| **Week 6:** | Maximal and Minimal Elements, Lattices |
| **Week 7:** | Foundations of Logic; Propositional Logics, System Specifications, Logic Puzzles |
| **Week 8:** | Predicates and Quantifiers – Predicates, Quantifiers, Universal Quantifiers, Quantifiers with Restricted Domains, Precedence, Binding Variables, Logical Equivalences, Negating Quantified Expressions, Translation to English Expressions |
| **Week 9:** | Rules of Inferences – Valid Arguments, Rules of Inferences for Propositional Logic, Building Arguments |
| **Week 10:** | Basic Counting Principles, Simple Counting Problems, Complex Counting Problems and Examples; The Inclusion-Exclusion Principle, Tree Diagrams; The Pigeonhole Principle; The Generalized PHP; Applications of Pigeonhole Principle; |
| **Week 11 :** | Permutations, r-Permutations, Combinations, r-Combinations, Binomial Coefficients, Examples and Applications; Binomial Coefficients, Binomial Theorem, Expression, and other Identities; Permutations and Combinations with Repetition, Permutations with Indistinguishable Objects, Distributing Events into Boxes; |

Generating Permutations and Combinations

**Week 12:** Graphs, Graph Models and Terminologies; Types of Graphs; Graph Isomorphism; Euler and Hamiltonian Paths and Circuits

**Week 13:** Spanning tree, Number of spanning tree of a graph, Some graph algorithms

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Professional and Business Communications

| | | |
|---|---|---|
| Course Code | : | **CSCC15** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | NIL |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course highlights the theory and practice of written, oral, and interpersonal communication used in the workplace with emphasis on composing clear, concise, and effective business correspondence. Students will discuss various types of communication media and the importance of succinct written and oral expression to modern business interactions. Students will have extensive practice writing a wide spectrum of documents, including professional email and reports technically using PC packages Word, Excel, Outlook express and Powerpoint. This course will emphasize vocabulary development, using correct grammar and punctuation, techniques for reducing writing time, and proofreading also are addressed. Students will develop and deliver an individual presentation, using appropriate and effective visual support, in which they present a persuasive argument that demonstrates relevance and benefits to an audience at different levels of expertise or interest. Further, because effective group communication is a necessity in today's workplace, students will learn and practice skills in low structure presentations, managing meetings, dealing with conflict, and leveraging the power of diversity, at both the individual and cultural level.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. To Use a strategic communication model and critical thinking to identify objectives, analyze audiences, and choose the most effective structure and style for delivering strategically sound written and spoken messages.
2. Practice principles of effective business writing and document design in all written documents by making technical use of PC packages Word, Excel and Powerpoint.
3. Design and deliver a persuasive presentation that convinces the audience of the topic's relevance and overcomes resistance, using appropriate visual support and adhering to a specified time limit.
4. Employ principles of effective group communication to cultivate trust and understanding, increase open participation, and strengthen decision making in work groups and teams.
5. Compose clear, concise, and effective business correspondence that meets the goal of various rhetorical situations: Letters, Cover letter, Letter for a job application, Thank you letter, Letter of complaint, Memos, Resumes, Emails, and Reports.
6. Understand and apply basic business etiquette to a variety of professional situations: First impressions, Attire, Listening skills, Tone, Behavior, Telephone etiquette.
7. As a team, design and deliver a presentation that both informs and persuades, using an appropriate visual support strategy and adhering to a specified time limit.

## UNITWISE SYLLABUS

1. **Professional and Business Communication:** Introduction to business communication, elements of communication process, Barriers to communication Basic types of business communication- Downward, Upward and Horizontal, Corporate communication network- Formal, Informal and Grapevine, Origin, Evolution and Growth of Corporate Communications, Tools of Corporate Communications, Key functions of Corporate Communications
2. **Practices in Business Communication:** English Grammar, Vocabulary development, Effective written and oral business communication, Writing Proposals, Memo, Business Letters, Request Letters, Sales Letters, Payment Collection Letters, Business Reports using technical features of PC packages such as Word, Excel, Outlook express, and Powerpoint, Ten Commandments of effective oral communication, Presentations, Group discussions, meetings.
3. **Non Verbal Professional Communication:** Body Language-Kinesics, Proxemics, Paralanguage, Interviews-types of interviews, preparing for interviews, appearing for interviews, interview process, Interviewing skills.
4. **Corporate Communications & Corporate Branding:** Introduction to Integrated Marketing Communications (IMC), IMC and Business Communications, Role of Corporate Communications in Creating and Managing Corporate Brands, Models for Managing Corporate Communications.

5. **Business Communications:** Trends and Issues, International Communication, Adapting to global business-negotiating skills and process, Collective bargaining process, Corporate Communications in the Digital Era, Ethics in Corporate Communications, Crisis Communications.

**Text Books**


1. Ronald E. Dulek , John S. Fielden : Principles of Business Communication, 1990, Prentice Hall.
2. Herta A. Murphy, Herbert William Hildebrandt: Business Communication, 7th Edition, The Mc Graw Hill.
3. Mary Ellen Guffey, Essentials of Business Communication, 8th Ed., 2010; Southwestern Cengage Learning ISBN: 978-1-439-05457-4
4. JoepCornilissen, 2004, Corporate Communications: Theory and Practice, Sage Publications.


**Reference Books**


1. Joseph Fernandez, 2004, Corporate Communication – A, 21st century Primer, Sage Publication.
2. Sandra M Oliver, 2004, Handbook of Corporate Communication and Public Relations, Routledge.
3. Lars Thoger Christensen, Mette Morsing and George Cheney, 2008, Corporate Communications: Convention, Complexity and Critique, Sage Publications.
4. William Strunk, Jr. and E.B. White; The Elements of Style, 4th Ed., 2000, ISBN:0-205-30902 X
5. The International Association of Business Communicators (IABC) http://www.iabc.com/


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction to business communication, elements of communication process, Barriers to communication Basic types of business communication-Downward, Upward and Horizontal |
| **Week 2:** | Corporate communication network- Formal, Informal and Grapevine, Origin, Evolution and Growth of Corporate Communications, |
| **Week 3:** | Tools of Corporate Communications, Key functions of Corporate Communications |
| **Week 4:** | English Grammar, Effective written and oral business communication, |
| **Week 5:** | Writing Proposals, Memo, Business Letters, Request Letters, Sales Letters |
| **Week 6:** | Payment Collection Letters, Business Reports, |

**Week 7:** Ten Commandments of effective oral communication, Presentations, Group discussions, meetings

**Week 8:** Interviews-types of interviews, preparing for interviews, appearing for interviews, interview process, Interviewing skills.

**Week 9:** Introduction to Integrated Marketing Communications (IMC), IMC and Business Communications,

**Week 10:** Role of Corporate Communications in Creating and Managing Corporate Brands

**Week 11:** Models for Managing Corporate Communications.

**Week 12:** Trends and Issues, International Communication, Adapting to global business-negotiating skills and process

**Week 13:** Collective bargaining process, Corporate Communications in the Digital Era, Ethics in Corporate Communications, Crisis Communications

**Week 14:** Revision

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

*********

# Course Plan and Detail Syllabi for

# Principles of Management and Organization Behaviour

| | | |
|---|---|---|
| Course Code | : | **CSCC16** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | NIL |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course provides a framework for understanding the day-to-day behaviour of employees within dynamic organizations. Topics covered include human behaviour principles in individual, small group, and organizational settings. The influence of job design on human performance will also be examined. The primary focus is on developing effective managerial skills used in improving employee performance. The overall goal is to present the student with the concepts and practices of organizational behaviour, which is the study of human behaviour, individual differences, and performance in organizational settings. The effective administration of an organization requires that its managers understand and use these organizational behaviour concepts and skills. Attention is given to the individual processes, group and interpersonal processes, and the organizational processes.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. The range of problems in the work place that can be solved through an understanding of employee behaviour.
2. The organizational culture and how it is controlled by both external forces (e.g., laws, litigation, federal guidelines) and internal components (e.g., employee characteristics, management style).

3. Principles of group interaction and its effect on organizational functioning, including both employee job satisfaction and productivity.
4. Different management styles and their effect on employee behaviour.
5. The multi-faceted nature of administrative effectiveness.

## UNITWISE SYLLABUS

1. **Introduction to Management:** Concept and Nature of management: Features, Levels of Management, Skills and role of managers, Significance and scope of management. Evolution of Management Thoughts: Approaches to management-Traditional and Contemporary, Management Process: Functions , Principles of Management, Forms of Business Ownerships
2. **Excellence in Management:** Modern Management Techniques: Benchmarking, Process Re-engineering, TQM, JIT, Competitive advantage, Organizational effectiveness: Effectiveness vs. Efficiency, McKinsey 7-S model, Concept and Significance of time management.
3. **Introduction to Organizational Behaviour**: Organizational Behaviour Concept, Models of OB, Contributing Disciplines to the OB, Challenges and Opportunities for Organizational Behaviour, Foundations of Individual Behaviour, Learning: Concept and theories of learning, reinforcement.
4. **Individual Processes:** Personality and Emotions, Perception and Attribution: Concept, nature and process, Employee Attitudes and Values, Motivation : Concepts and their applications, Theories of Motivation
5. **Management, Environment & Strategy:** Components of Business Environment; Impact of Environment; Organization and environment interface; Corporate Strategy: Concept; Nature; Levels, Strategic Planning; Strategic Change. to business communication, elements of communication process, Barriers to communication, Basic types of business communication-Downward, Upward and Horizontal, Corporate communication network-Formal, Informal and Grapevine, Origin, Evolution and Growth of Corporate Communications, Tools of Corporate Communications, Key functions of Corporate Communications.

### Text Books


1. Gupta. C.B, Business Organization and Management, Mayur Paperbacks.
2. Gilbert, Principles of Management, Mc Graw Hill.
3. Y.K. Bhushan, Fundamentals of Business Organization and Management.
4. Kaul Vijay Kumar, Business Organization and Management: Text and Cases, Pearson.


### Reference Books


1. Greenberg Jerald and Baron Robert A, Behavior in Organizations.
2. Richard. I. Daft, Principles of Management, Cengage Learning India.

3. Singh Kavita, Organization Behavior, Vikas Publications.
4. Luthans Fred, Organizational Behavior, Tata McGraw Hill.
5. Newstrom John W, Organizational Behavior, Tata McGraw Hill.
6. Newstrom, J. W., & Davis, K. (2002). Organizational behavior: Human behavior at work (11th ed.). New York: McGraw-Hill. (ISBN 0-07-287228-4)

### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Concept and Nature of management: Features, Levels of Management, Skills and role of managers, Significance and scope of management |
| **Week 2:** | Evolution of Management Thoughts: Approaches to management-Traditional and Contemporary |
| **Week 3:** | Management Process: Functions , Principles of Management, Forms of Business Ownerships |
| **Week 4:** | Modern Management Techniques: Benchmarking, Process Re-engineering |
| **Week 5:** | TQM, JIT, Competitive advantage, Organizational effectiveness: Effectiveness vs. Efficiency |
| **Week 6:** | McKinsey 7-S model, Concept and Significance of time management |
| **Week 7:** | Organizational Behavior Concept, Models of OB |
| **Week 8:** | Contributing Disciplines to the OB, Challenges and Opportunities for Organizational Behavior |
| **Week 9:** | Foundations of Individual Behavior, Learning: Concept and theories of learning, reinforcement |
| **Week 10:** | Personality and Emotions, Perception and Attribution: Concept, nature and process |
| **Week 11:** | Employee Attitudes and Values, Motivation : Concepts and their applications, Theories of Motivation |
| **Week 12:** | Components of Business Environment; Impact of Environment; Organization and environment interface, Corporate Strategy: Concept; Nature; Levels, Strategic Planning; Strategic Change. to business communication, elements of communication |

process, Barriers to communication

**Week 13:** Basic types of business communication-Downward, Upward and Horizontal, Corporate communication network-Formal, Informal and Grapevine, Origin, Evolution and Growth of Corporate Communications, Tools of Corporate Communications, Key functions of Corporate Communications

**Week 14:** Revision

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Operating System and Shell Programming

| | | |
|---|---|---|
| Course Code | : | **CBCS21** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11 |
| Course Status | : | CBCS |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:**

Concept of Operating System and Shell Programming are necessary to demonstrate knowledge of process control, threads, concurrency, memory management scheduling, I/O and files, distributed systems, security, networking. Student teams will implement a significant portion of an operating System.

**b) Objectives**

At the end of the course, the students will be prepared to:

1. Demonstrate an ability to analyze a problem and identify and define the computing requirements appropriate to its solution.
2. Understand and analyse theory and implementation of: processes, resource control (concurrency etc.), physical and virtual memory, scheduling, I/O and files.
3. Understand the high-level structure of the Linux kernel both in concept and source code.
4. Acquire a detailed understanding of one aspect (the scheduler) of the Linux kernel as well as shell Programming.

## UNITWISE SYLLABUS

1. **Introduction:** Definition and types of operating systems, Batch Systems, multi programming, time–sharing parallel, distributed and real-time systems, Operating system structure, Operating system components and services, System calls, system programs, Virtual machines.
2. **Process Management, Synchronization and Deadlocks:** Process concept, Process scheduling, Cooperating processes, Threads, Inter process communication, CPU scheduling criteria, Scheduling algorithms, Multiple-processor scheduling, Real-time scheduling and Algorithm evaluation. The Critical-Section problem, synchronization hardware, Semaphores, Classical problems of synchronization, Critical regions, Monitors, Deadlocks-System model, Characterization, Deadlock prevention, Avoidance and Detection, Recovery from deadlock, Combined approach to deadlock handling.
3. **Storage management:** Memory Management-Logical and Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Virtual Memory, Demand paging and its performance, Page replacement algorithms, Allocation of frames, Thrashing, Page Size and other considerations, Demand segmentation, secondary Storage Structure, File concept, access methods, directory implementation, Efficiency and performance, recovery, Disk structure, Disk scheduling methods, Disk management, Recovery, Disk structure, disk scheduling methods, Disk management, Swap-Space management, Disk reliability.
4. **Threats and Security:** Protection and Security-Goals of protection, Domain of protection, Access matrix, Implementation of access Matrix, Revocation of Access Rights, language based protection, The Security problem, Authentication, One Time passwords, Program threats, System threats, Threat Monitoring, Network structure security. Windows NT-Design principles, System components, Environmental subsystems, File concept, File system, Networking and program interface, Linux system-design principles, Kernel Modules, Process Management
5. **Concepts of Shell Programming:** Types of shells, Shell functionality, Environment, Writing script & executing basic script, Debugging script, Making interactive scripts, Variables (default variables), Functions & file manipulations, Processing file line by line, Functions, Regular Expression & Filters. Advanced Scripting Techniques: Providing command line options to scripts, Shell & sub shells, Exporting variables, Remote shell execution, Dialog boxes, SQL with Shell, Connecting to MySQL using shell, Running SQL queries from a shell script

**Text Books**

1. Abraham Siberschatz and Peter Baer Galvin, “Operating System Concepts”, Addision-Wesley.
2. D M Dhamdhere, “Operating Systems : A Concept basedApproach”, McGraw Hill.
3. Harvey M Deital, "Operating Systems", Addison Wesley.
4. Milan Milankovic, “Operating Systems, Concepts and Design”, Tata McGraw-Hill.

**Reference Books**

1. Richard Peterson, “Linux: The Complete Reference”, Osborne Tata McGraw-Hill.
2. SibsankarHalder and Alex A Aravind, “Operating Systems”, Pearson Education.
3. Charles Crowley, “Operating Systems: A Design-Oriented Approach”, Tata McGraw Hill Education”.
4. Stuart E. Madnick & John J. Donovan. Operating Systems. McGraw Hill.

**c) Outline**

| **Weeks:** | **Topics:** |
|---|---|
| Week -1 | Definition and types of Operating Systems. |
| Week -2 | Operating system structure. |
| Week- 3 | Process Management. |
| Week- 4 | CPU scheduling criteria, Scheduling algorithms.. |
| Week- 5 | Process Synchronization and Deadlocks. |
| Week -6 | Deadlocks-System model, Characterization, Deadlock prevention. |
| Week -7 | Memory Management-Logical and Physical Address Space. |
| Week -8 | Allocation of frames, Thrashing, Page Size and other considerations. |
| Weeks- 9 | Protection and Security-Goals of protection, Domain of protection. |
| Week -10 | Program threats, System threats, Threat Monitoring,. |
| Week -11 | File concept, File system, Networking and program interface, |
| Week -12 | Linux system-design principles, Kernel Modules. |
| Week -13 | Concepts of Shell Programming. |
| Week -14 | Advanced Scripting Techniques. |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Object Oriented Programming in C++

| | | |
|---|---|---|
| Course Code | : | **CSCC622** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Object Oriented Programming (OOP) is a programming paradigm organized around objects rather than actions, and data rather than logic. It is one of the most widely used programming paradigms used in software industries today. This course has been designed to give the fundamental concepts of OOP, including abstraction, data hiding, inheritance, polymorphism, dynamic method dispatch, generic programming and stream. Various C++ language features will be covered, like Dynamic initialization, Classes and objects, Function prototyping, Array of objects, Constructors, Operator overloading, Inheritance, Templates, Streams, Exception Handling. Object Oriented Methodologies.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Design classes for a particular problem, and achieve data abstraction and hiding.
2. Define constructions, destructors and create objects of class.
3. Understand the purpose of inheritance and apply it to real-life problem solving.
4. Understand the concept of Polymorphism and apply it to real-life problem solving.
5. Write generic functions and classes.
6. Understand and apply the concept of data and byte streaming for File handling.

## UNITWISE SYLLABUS

1. **Introduction**: Programming Paradigms: Unstructured Programming, Structured Programming, Object Oriented Programming; Abstract Data Type (ADT); Class; Object; Message; Encapsulation; Polymorphism; Inheritance; Pros and Cons of Object Oriented Methodology; cin and cout Objects.
2. **Classes and Objects:** Classes; Friend Functions: Introduction, Benefits and Restrictions, Friends Classes; Inline Functions; Constructor: Introduction, Parameterized Constructor; Destructor and its usages; Static Data Member and Static Member Functions; Creating Object; Passing and Returning Object(s) to/from a Function; Object Assignment; Nested and Local Classes; Arrays of Objects; Pointer to Objects; this Pointer, Pointer to Derived Type; References; Reference vs Pointer; Reference Parameters; Dynamic Memory Allocation: new and delete Operators.
3. **Function and Operator overloading**: Function overloading: Introduction, Rules, Overloading Constructors, Copy Constructors; Default Function Arguments vs. Function Overloading. Operator Overloading: Introduction, Operators that cannot be Overloaded, Overloading Operators using Member Function and Friends Functions, Overloading different operators including prefix and postfix form of ++ and -- operators, Shorthand Operators, new, delete, [], (), -> and comma Operators.
4. **Inheritance & Virtual function**: Inheritance: Introduction, Types of Inheritances, Base-Class Access Control, Protected Members, Protected Base-class Inheritance, Multiple Inheritance, Problem in Multiple Inheritance, Solution to Multiple Inheritance Problem, Passing Parameters to Base Class Constructors; Virtual functions: Introduction, Calling a Virtual Function using Base Class Reference, Pure Virtual Function, Abstract Class.
5. **Generic Function, Exception and File Handling:** Generic Functions: Benefits, Functions with Two Generic Types, Explicitly Overloading a Generic Function, Overloading a Function Template, Restriction, Generic Sort, Generic Class. Exception Handling: Introduction, Using try and catch Blocks, Creating Exception Class, throwing Object. C++ Streams; C++ File Handling: Opening and Closing a File, Reading and Writing a Text File, Random Access, Reading and Writing Object to a File

### Text Books


1. Herbert Schildt: Complete Reference C++, 3rd Ed., Tata Mc Graw Hill
2. BJarne Stroustrup: The C++ Programming Language, Pearson Education
3. E. Balagurusamy: Object Oriented Programming with C++ 6th Edition , Mc Graw Hill.
4. H.M. Deitel & P.J. Deitel: C++ How to Program, 4th Ed., Pearson Education.

**Reference Books**


1. Askok N. Kamthane: Object Oriented Programming with ANSI & TURBO C++, Pearson Education.
2. Bruce Eckel: Thinking in C++, Pearson Education
3. N.Barkakati: Object Oriented Programming in C++, PHI
4. K.R. Venugopal & Rajkumar Buyya: Mastering C++ by all, 2nd Ed., Tata McGraw Hill


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Programming Paradigms: Unstructured Programming, Structured Programming, Object Oriented Programming; Abstract Data Type (ADT). |
| **Week 2:** | Class; Object; Message; Encapsulation; Polymorphism; Inheritance; Pros and Cons of Object Oriented Methodology; cin and cout Objects. |
| **Week 3:** | Classes; Friend Functions: Introduction, Benefits and Restrictions, Friends Classes; Inline Functions; Constructor: Introduction, Parameterized Constructor; Destructor and its usages. |
| **Week 4:** | Static Data Member and Static Member Functions; Creating Object; Passing and Returning Object(s) to/from a Function; Object Assignment; Nested and Local Classes. |
| **Week 5:** | Arrays of Objects; Pointer to Objects; this Pointer, Pointer to Derived Type; References; Reference vs Pointer; Reference Parameters; Dynamic Memory Allocation: new and delete Operators. |
| **Week 6:** | Function overloading: Introduction, Rules, Overloading Constructors, Copy Constructors; Default Function Arguments vs. Function Overloading. |
| **Week 7:** | Operator Overloading: Introduction, Operators that cannot be Overloaded, Overloading Operators using Member Function and Friends Functions. |
| **Week 8:** | Overloading different operators including prefix and postfix form of ++ and -- operators, Shorthand Operators, new, delete, [], (), -> and comma Operators. |
| **Week 9:** | Inheritance: Introduction, Types of Inheritances, Base-Class Access Control, Protected Members, Protected Base-class Inheritance. |
| **Week 10:** | Multiple Inheritance, Problem in Multiple Inheritance, Solution to Multiple Inheritance Problem, Passing Parameters to Base Class Constructors. |

**Week 11:** Virtual functions: Introduction, Calling a Virtual Function using Base Class Reference, Pure Virtual Function, Abstract Class.

**Week 12:** Generic Functions: Benefits, Functions with Two Generic Types, Explicitly Overloading a Generic Function, Overloading a Function Template, Restriction, Generic Sort, Generic Class, Exception Handling: Introduction, Using try and catch Blocks, Creating Exception Class, throwing Object.

**Week 13:** C++ Streams; C++ File Handling: Opening and Closing a File, Reading and Writing a Text File, Random Access, Reading and Writing Object to a File.

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# DATA & FILE STRUCTURE

| | | |
|---|---|---|
| Course Code | : | **CSCC23** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:**

Introductory concept of Data File Structure are necessary to Analyses and implement the advanced data structures such as balanced search trees, hash tables, priority queues and the disjoint set union/find data structure .It is also necessary to be familiar with several sub-quadratic sorting algorithms including quick sort, merge sort and heap sort .Also useful with some graph algorithms such as shortest path and minimum spanning tree the implementation of linked data structures such as linked lists and binary trees.

**b) Objectives**

At the end of the course, the students will be prepared to:

1. Understand the abstract data types stack, queue, deque, and Linked List.
2. Understand prefix, infix, and postfix expression formats and use stacks to evaluate postfix expressions.
3. be able to recognize problem properties where stacks, queues, and deques are appropriate data structure
4. Able to compare the performance of Various Sorting algorithms and our linked list implementation.

## UNITWISE SYLLABUS

1. **List and Matrices:** Data Structure, Linear Data Structure, Array, Address Calculation, Single Linked List, Circular Linked List, Doubly Linked List, Circular Doubly Linked List, Applications of Arrays and Linked List, Matrix, Mapping of Matrix elements to One Dimensional (1D) Array, Special Matrices, Triangular, Diagonal, Tri-Diagonal, Representation in Row Major and Column Major Order, Mapping of non-null Elements in 1D Array, Sparse Matrix, Applications of Linked Lists: Bin Sort, Radix Sort, Convex Hull.
2. **Stack and Queues:** Stack Data Structure, Push & Pop Operations, Representation and Implementation of Stack using Array and Linked List, Applications of Stack: Conversion of Infix to Postfix Expressions, Parenthesis Matching, Towers of Hanoi, Rat in a Maze, Implementation of Recursive Functions, Queue Data Structure, Various Queue Operations, Circular Queue, Representation and implementation of queues using Array and Linked List, Deque, Applications of Queue Railroad Car Rearrangement, Machine Shop Simulation, Image-Component Labeling, Priority Queues: Priority Queue Using Heap; Max and Min Heap; Insertion into Heap; Deletion from a Heap; Applications of Priority Queue: Heap Sort, Huffman Codes.
3. **Trees**: Basic Concepts, Binary Trees and their Properties; Representation of Binary Trees: Array-Based and Linked Representations; Binary Tree Traversals; Binary Search Trees (BST); Operations on BST: Search, Insertion and Deletion; BST with Duplicates; Indexed BST; Applications of BST: Histogramming, Best-Fit Bin Packing. B-Trees and their Representation; Operations on B-Tree: Search, Insertion and Deletion; B+-Trees; AVL Trees; AVL Tree Representation; Operations on AVL Trees: Search, Insertion and Deletion; Introduction to Red-Black and Splay Trees.
4. **Sorting and Searching**: Insertion Sort, Bubble Sorting, Quick Sort, Merge Sort, Shell sort, Sequential search, binary search, Introduction to Hashing, Hash Table Representation, Hash Functions, Collision and Overflows, Linear Probing, Random Probing, Double Hashing, and Open Hashing.
5. **Graphs and File Structure:** Terminology & Representations, Graphs & Multi-graphs, Directed Graphs, Representations of Graphs, Weighted Graph

Representations; Graph Traversal Methods: Breadth-First Search and Depth-First Search; Spanning Tree and Shortest Path Finding Problems, Physical Storage Media File Organization, Organization of records into Blocks, Sequential Files, Indexing and Hashing, Primary indices, Secondary indices, B+ Tree index Files, B Tree index Files, Indexing and Hashing Comparisons.

**Text Books**


1. Y. Langsam, M. Augenstin and A. Tannenbaum, Data Structures using C and C++, Pearson Education Asia, 2nd Edition, 2002
2. Sahni: Data Structures, Algorithms and Applications, Galgotia.
3. S. Lipschutz, Data Structures Mc-Graw Hill International Editions, 1986.


**Reference Books**


1. Jean-Paul Tremblay, Paul. G. Soresan, An introduction to data structures with Applications, Tata Mc-Graw Hill International Editions, 2nd edition 1984.
2. A. Michael Berman, Data structures via C++, Oxford University Press, 2002.
3. M. Weiss, Data Structures and Algorithm Analysis in C++, Pearson Education, 2002, 2nd edition.
4. D. Samanta, Classic Data Structure.


**c) Outline**

| **Weeks:** | **Topics:** |
|---|---|
| Week 1: | Basic Terminology of Data Structure. |
| Week 2: | Representation and Implementation of Array & Stack. |
| Week 3: | Conversion of Infix to Prefix and Postfix Expressions. |
| Week 4: | Representation and implementation of queues. |
| Week 5: | Representation and Implementation of Linked List & their types. |
| Week 6: | Tower of Hanoi Problem, Backtracking, recursive algorithms. |
| Week 7: | Basic terminology of Binary Trees. |
| Week 8 : | Binary Search Tree (BST), Insertion and Deletion in BST. |
| Weeks 9: | Searching and Hashing. |

| | |
|---|---|
| Week 10: | Various Sorting Techniques.. |
| Week 11 : | Terminology & Representations of Graph & their types. |
| Week 12: | Spanning Trees, Minimum Cost Spanning Trees. |
| Week 13: | Physical Storage Media & File Organization, Indexing and Hashing Comparisons. |
| Week 14 : | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Microprocessor & Computer Architecture

| | | |
|---|---|---|
| Course Code | : | **CSCC24** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12/csss13 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course is an introduction to computer architecture, in which the basic architecture of computer shall be described. The details of CPU and computer arithmetic will be the core areas to focus on. Besides this, machine programming, memory organization and I/O organization will also be covered in this course.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Understand basic architecture of computer.
2. Use machine programming like machine 0 to machine 3.
3. Understand the concepts of I/O and memory organization.
4. Gain awareness about parallel processing, vector processing and array processing.

## UNITWISE SYLLABUS

1 **Microprocessor:** Microprocessor, Survey of Microprocessor, Evolution of Microprocessor, Microprocessor Generations, Microprocessor Architecture, Architecture Tree, Functional Block Diagram of 8085, Register Section, Timing and Control Unit, Addressing Modes (General).

2 **Assembly Programming 8085:** Addressing Modes of 8085, Instruction Sets, Instruction

Classification, Instruction Format, Data Format, Opcode Format, Writing Assembly Programs, Timing Diagram and Machine Cycle.

3 **Input-Output Organization:** Peripheral Devices; Input Output Interface; I/O vs. Memory Bus; Isolated vs. Memory Mapped I/O; Asynchronous Data Transfer – Strobe Control, Handshaking; Mode of Data Transfer; Priority Interrupt – Daisy Chaining Priority, Parallel Priority Interrupt, Priority Encoder, Interrupt Cycle, Software Routines; Direct Memory Access (DMA) – DMA Controller, DMA Transfer; Input-Output Processor (IOP), Architecture of 8086, Register set of 8086, 8086 Operational Modes.

4 **Central Processing Unit (CPU) and Some Arithmetic Operations:** Introduction; General Register Organization; Control Word; Stack Organization – Register Stack, Memory Stack, Reverse Polish Notation, Evaluation of Arithmetic Expression. Instruction Format – Three Address Instructions, Two Address Instructions, One Address Instructions, Zero Address Instructions. Addressing Modes; Program Control - Status Bit Conditions, Conditional Branch Instructions, Subroutine Call; Program Interrupt – Types of Interrupts; RISC and CISC Characteristics, Addition, Subtraction, Booth's Multiplication, Division Algorithm.

5 **Memory Organization and Pipeline:** Memory Hierarchy; Associative Memory – Hardware Organization, Match Logic, Read Operation, Write Operation; Cache Memory – Associative Mapping, Direct Mapping, Set-Associative Mapping, Writing into Cache, Cache Initialization, Parallel Processing; Pipelining – Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline; Vector Processing – Vector Operations, Matrix Multiplication, Memory Interleaving; Array Processor – Attached Array Processor, SIMD Array Processor.

**Text Books**


1. M. Morris Mano: Computer System Architecture, 3rd Ed., PHI
2. Gaonkar: Microprocessor Architecture, Programming and Applications with 8085, PRI
3. D.V. Hall: Microprocessing and Interfacing, TMH
4. B. Ram: Fundamental of Microprocessor and Micro-Controller, DRP.
5. Willian Stallings: Computer Organization and Architecture: Design for Performance, Pearson.


**Reference Books**


1. Rafiquzzaman & Chandra: Modern Computer Architecture, 2003.
2. William Stalling,: Computer Organization and Architecture, Pearson Education
3. Rajaraman & T. Radhakrishnan: Computer Organization and Architecture, PHI.

**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Register Transfer and Micro Operations: Register Transfer Language (RTL); Register Transfer; Bus Transfer; Memory Transfers |
| **Week 2:** | Arithmetic Microoperations; Logic Microoperations – List of Logic Microoperations, Hardware Implementation, Some Applications; Shift Microoperations; Arithmetic Logic Shift Unit |
| **Week 3:** | Introduction; General Register Organization; Control Word; Stack Organization – Register Stack, Memory Stack, Reverse Polish Notation, Evaluation of Arithmetic Expression. |
| **Week 4:** | . Instruction Format – Three Address Instructions, Two Address Instructions, One Address Instructions, Zero Address Instructions. Addressing Modes; |
| **Week 5:** | Program Control - Status Bit Conditions, Conditional Branch Instructions, Subroutine Call; Program Interrupt – Types of Interrupts; RISC and CISC Characteristics |
| **Week 6:** | Binary Arithmetic Operations**:** Addition and Subtraction – with Signed Magnitude Data; Multiplication – Algorithms, |
| **Week 7:** | Booth Multiplication Algorithm; Division Algorithm – Divide Overflow; Floating Point Arithmetic Operations – Addition, Subtraction,. |
| **Week 8:** | Multiplication, Division; Decimal Arithmetic Operations – Addition, Subtraction, Multiplication, and Division |
| **Week 9:** | Peripheral Devices; Input Output Interface; I/o vs. Memory Bus; Isolated vs. Memory Mapped I/O; Asynchronous Data Transfer – Strobe Control, Handshaking; |
| **Week 10:** | Mode of Data Transfer; Priority Interrupt – Daisy Chaining Priority, Parallel Priority Interrupt, Priority Encoder, Interrupt Cycle, Software Routines; Direct Memory Access (DMA) – DMA Controller, DMA Transfer; Input-Output Processor (IOP) |
| **Week 11:** | **:** Memory Hierarchy; Main Memory – RAM, ROM, Memory Address Map, Memory Connection to CPU; Auxiliary Memory – Magnetic Disks, Magnetic Tape; Associative Memory – Hardware Organization, Protection. |
| **Week 12:** | Match Logic, Read Operation, Write Operation; Cache Memory – Associative Mapping, Direct Mapping, Set-Associative Mapping, Writing into Cache, Cache |

Initialization; Virtual Memory – Address Space and Memory Space, Address Mapping using Page, Associative Memory Page Table, Page Replacement; Memory Management Hardware – Segmented Page Mapping, Memory

**Week 13:** Parallel Processing; Pipelining – Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing – Vector Operations, Matrix Multiplication, Memory Interleaving; Array Processor – Attached Array Processor, SIMD Array Processor

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# System Analysis and Design

| | | |
|---|---|---|
| Course Code | : | **CSCC25** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course will teach the students to effectively analyze and design a software system. It will include various tools for analysis of structured and object-oriented systems. Tools like DFD, use case diagrams and models will be the focus of study. The course will include various real life case studies from the University set up to explain the concepts in great detail.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Use Ganntt and pert chart for effective scheduling of a software project.
2. Use DFD as a tool for detailed analysis of any given system.
3. Use use-case diagrams for modeling object-oriented systems.
4. Understand role of the key stake holders in any software project with the role of systems analyst in detail.
5. Communicate scientific knowledge at different levels of abstraction.

## UNITWISE SYLLABUS

1. **System and System Analyst:** System Concepts; Information and Management; Information Architecture in the Organizations; Management Triangle; Computer-Based Information System – Need & Significance; Types of Information System, Factors Affecting the Information Systems; Drivers of Information system; System Stakeholders System Analyst – Roles and Responsibilities, Skills and Characteristics.

2. **System Development Life Cycle:** Methodologies of System Development, System Analysis – Identifying Problems and Objectives, Systems Proposal, Questionnaires, Interviewing, Brain-Storming, Prototyping, Determining Information Requirements, Analyzing System Needs, System Specifications, Ascertaining Hardware and Software Needs and Selection; Feasibility Study; Cost-Benefit Analysis; Planning and Scheduling of Projects.

3. **Structured Analysis:** Tools used for Structured Analysis, Data Flow Diagram (DFD), Logical and Physical DFDs, Data Dictionary and Process Specification, Structured English, Decision Tables, Decision Trees.

4. **Systems Design:** Designing the Recommended System, Structured Design, Tools for Structured Design, Modular Decomposition, Top-Down and Bottom-Up Designs, Using Structure Charts to Design Systems, Designing Effective Input, Designing Effective Output, Form Design, Designing Databases, Designing User-Interfaces.

5. **Object-Oriented Analysis:** Software Complexity; S/W Crisis & Related Issues; Object-Oriented Approach, Features & Significance; Object Oriented Methodologies; Modeling Concepts; Object Modeling – Objects and Classes, Links & Associations, Generalization & Inheritance, Grouping Constructs, Aggregation, Abstract Classes, Multiple Inheritance, Meta Data, Candidate Keys and Constraints. Dynamic Modeling – Events and States, Operations; Functional Modeling – DFDs, Specifying Operations and Constraints.

### Text Books


1 J. L. Witten & L. D. Bentley: Systems Analysis and Design Methods, 7th Ed., McGraw-Hill

2 Kendall & Kendall: Systems Analysis and Design, 5th Ed., Pearson Education

3 Roger S. Pressman: Software Engineering: A Practitioner's Approach, 6th Edition Mc Graw Hill Publication.

4 Ian sommerville:, Software Engineering, 9th Edition, 2010, Pearson Education.

## Reference Books


1 Rambaugh, Blaha & Eddy: Object-Oriented Modeling and Design, 12th Ed., PHI.

2 Grady Booch: Object-Oriented Analysis and Design, 2nd Ed., Pearson Education

3 Hans Van Vliet: Software Engineering: Principles and Practices–, 2008.

4 Richard Fairley: Software Engineering Concepts, 2008.


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction to computer based Information Systems. |
| **Week 2:** | Drivers of information systems the various stake holders and role of the Systems Analyst. |
| **Week 3:** | Systems analysis techniques. Details of questionnaires, interviews, etc. |
| **Week 4:** | Feasibility analysis of a new system along with cost-benefit analysis. |
| **Week 5:** | Various models for systems development. |
| **Week 6:** | DFD as an effective tool for analysis. |
| **Week 7:** | Data dictionary and decision table modeling. |
| **Week 8:** | Structured design concept |
| **Week 9:** | Database design with introduction to table design |
| **Week 10:** | Structure charts and top down plus bottom up design |
| **Week 11:** | Object oriented design concept |
| **Week 12:** | Use case modeling |
| **Week 13:** | Objects and classes, abstract class design |
| **Week 14:** | Revision |

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Theory of Computations

| | | |
|---|---|---|
| Course Code | : | **CSCC26** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11, CBSE12, CSCC14, CSCC26 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This is the study of abstract computing device. Finite automata, a simple machine, are proposed to model brain function. The formal grammars have close relationship with abstract automata and serve as a basis of some important software components, including part of compilers. This course covers finite automata, regular expression, context free grammars and languages, pushdown automata, and Turing machine.

**b) Objectives:** Upon completion of this course, students will be able to do the following:

1. Understand the finite automata like DFA, and NFA and conversion from NFA to DFA.
2. To define some formal languages.
3. To write the regular expression for some formal languages.
4. To write the grammars for some languages, and generating the parse tree for given strings.
5. To describe the languages that cannot be modeled using finite automata.

## UNITWISE SYLLABUS

1. **Introduction:** Finite Automata, Formal Proofs, Deductive Proofs, Contrapositive, Proof by Contradiction, Proof by Counter Example, Proof by Induction, Concept of Automata Theory, Finite Automata: Deterministic Finite Automata (DFA), Languages of DFA; Non Deterministic Finite Automata (NFA), Language of NFA, Equivalence of Deterministic and Non-deterministic Automata, Application of Automata: Finding String in Text, Recognizing a Set of Keywords, Finite Automata with Epsilon Transition.
2. **Regular Languages and Regular Grammars:** Regular Expressions, Finite Automata and Regular Expressions, Conversion from DFA to Regular Expression, Conversion from Regular Expression to Automata, Languages Associated with Regular Expressions, Connection between Regular Expressions and Regular languages, Regular Grammar, Properties of Regular Languages, Closure properties of Regular Languages. Identifying Non-regular Languages**.**
3. **Context Free Languages:** Context Free Grammars, Examples of Context Free Languages, Left most and Right most Derivations, Derivation Trees, Relationship between Derivation and Derivation Trees, Ambiguity in Grammars and Languages, Ambiguous Grammar, Methods for transforming Grammars; An useful Substitution Rule, Removing Useless productions, Removing λ- productions, Removing unit productions, Two important Normal Forms: Chomsky Normal Forms and Greibach Normal Form; Pumping Lemma for CFLs.
4. **Pushdown Automata:** Push Down Automata (PDA), Informal and Formal Definition of a Push Down Automata, Descriptions of a PDA, the Language Accepted by a Push Down Automata, Push Down Automata and Context Free Languages, Context Free Grammar for Push Down Automata; Deterministic Push Down Automata.
5. **Turing Machines and their Languages:** The Standard Turing Machine, Definition of a Turing Machine, Turing Machine Language Accepters, Other Models of the Turing Machine, Multi-tape Turing Machines, Multidimensional Turing Machines, Nondeterministic Turing Machines, The Universal Turing Machine; Recursive and Recursively Enumerable languages, Some Problems that cannot be Solved by Turing Machines- Computability and Decidability.

### Text Books


1. Linz: Theory of Automata and Formal Languages, JBL.
2. Hopcroft et al: Introduction to Automata Theory, Languages and Computation, Pearson.
3. J. Hopcroft, R. Motwani, and J. Ullman: Introduction to Automata Theory, Languages, and Computation, 3rd edition, 2007, Pearson/Addison-Wesley.
4. Daniel I.A. Cohen: Introduction to Computer Theory, John Wiley.

**Reference Books**


1. Mishra: Theory of Computer Science, PHI
2. Lewis & Papadimitriou: Elements of the Theory of Computation, PHI
3. John C Martin, Introduction to Languages and The Theory of Computation, McGraw Hill.
4. Michael Sipser: Introduction to the Theory of Computation, 3rd edition, 2013, Cengage Learning.


**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Finite Automata, Formal Proofs, Deductive Proofs, Contrapositive, Proof by Contradiction, Proof by Counter Example |
| **Week 2:** | Proof by Induction, Concept of Automata Theory, Finite Automata: Deterministic Finite Automata (DFA), Languages of DFA |
| **Week 3:** | Non Deterministic Finite Automata (NFA), Language of NFA, Equivalence of Deterministic and Non-deterministic Automata, Application of Automata: Finding String in Text, Recognizing a Set of Keywords, Finite Automata with Epsilon Transition |
| **Week 4:** | Regular Expressions, Finite Automata and Regular Expressions, Conversion from DFA to Regular Expression, Conversion from Regular Expression to Automata, Languages Associated with Regular Expressions |
| **Week 5:** | Connection between Regular Expressions and Regular languages, Regular Grammar, Properties of Regular Languages, Closure properties of Regular Languages. Identifying Non-regular Languages |
| **Week 6:** | Context Free Grammars, Examples of Context Free Languages, Left most and Right most Derivations, Derivation Trees, Relationship between Derivation and Derivation Trees |
| **Week 7:** | Ambiguity in Grammars and Languages, Ambiguous Grammar, Methods for transforming Grammars; An useful Substitution Rule, Removing Useless productions, Removing λ- productions, Removing unit productions, Two important Normal Forms: Chomsky Normal Forms and Greibach Normal Form; Pumping Lemma for CFLs. |
| **Week 8:** | Push Down Automata (PDA), Informal and Formal Definition of a Push Down Automata, Descriptions of a PDA, the Language Accepted by a Push Down Automata |
| **Week 9:** | Push Down Automata and Context Free Languages, Context Free |

| | |
|---|---|
| | Grammar for Push Down Automata; Deterministic Push Down Automata |
| **Week 10:** | The Standard Turing Machine, Definition of a Turing Machine, Turing Machine Language Accepters, Other Models of the Turing Machine |
| **Week 11 :** | Multi-tape Turing Machines, Multidimensional Turing Machines, Nondeterministic Turing Machines, The Universal Turing Machine |
| **Week 12:** | Recursive and Recursively Enumerable languages, Some Problems that cannot be Solved by Turing Machines- Computability and Decidability |
| **Week 13:** | Revision |
| **Week 14:** | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Advance Problem Solving using Java Programming

| | | |
|---|---|---|
| Course Code | : | **CSCC31/CBCS31** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC11/CBCS11/CSCC12/CBSC12/CSCC21/CSCC22 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Java is a general-purpose, object-oriented, powerful and widely used programming language that has been equipped with many modern programming features including class-based, object-oriented, structured, imperative, generic, reflective, concurrent, automatic garbage collected, and architecture neutral. As of 2016, Java has been ranked as mostly used programming language, especially for web-based application, with 9 million developers. The main principles on which Java language is created, are: simple, object-oriented, familiar, robust and secure, architecture-neutral and portable, high performance, interpreted, threaded, and dynamic.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Understand fundamentals of Java programming constructs.
2. Design classes, objects and apply code reusability using inheritance.
3. Handle String by using powerful String classes such as StringBuffer, StringBuilder and StringTokenizer.
4. Write generic, multithreaded, GUI-based programming.
5. Handle exceptions and write programs that needs database handling.

## UNITWISE SYLLABUS

1. **Introduction, Environment and Programming Structure:** Java White Paper Buzzwords, History of Java, Common Misconceptions, Choosing a Development Environment: Command-Line Tools, Running a Graphical Application, Building and Running Applets; A Simple Java Program, Comments, Data Types, Variables, Operators, Input and Output, Control Flow, Big Numbers, Arrays.
2. **Class, Objects and Inheritance:** Introduction to OOP, Predefined Classes, User Defined Classes, Static Fields and Methods, Method Parameters, Object Construction, Packages, Class Path, Documentation Comments, Class Design; Inheritance: Super-classes and Subclasses, Types of Inheritance, Polymorphism, Abstract class, Object: The Cosmic Super class, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration Classes, Reflection, Inheritance Guidelines, Interfaces.
3. **String Handling, Exception Handling and Generic Programming:** String Handling APIs: String, Immutable String, Methods of String Class, StringBuffer, StringBuilder, StringTokenizer. Exceptions: Dealing with Errors, Catching Exceptions, Guidelines for Using Exceptions, Assertions, Logging; Generic Programming: Definition, Generic Methods, Bounds for Type Variables, Generic Code and VM, Restrictions and Limitations, Inheritance Rules for Generic Types, Reflection and Generics.
4. **Java Collections and Multithreading:** Collection Interfaces, Concrete Collections, The Collections Framework, Algorithms, Legacy Collections, Multithreading: Threads, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues, Thread-Safe Collections, Callable and Futures, Executors, Synchronizers.
5. **Java GUI Programming and JDBC:** Introduction to Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Displaying Images, Event Handling, Basics of Event Handling, Actions, Mouse Events, The AWT Event Hierarchy; JDBC: Basic JDBC Programming Concepts, JDBC Drivers, Statements, Executing Queries, Result Sets.

### Text Books


1. Horstmann & Cornell: Core Java Volume I: Fundamentals, Pearson Education
2. Horstmann & Cornell: Core Java Volume II: Advanced Features, Pearson Education.
3. H. Schildt: Java 2: The Complete Reference (9th Ed.), 2014, Tata McGraw Hill.
4. Daniel Liang: Introduction to Java programming, 7th Edition Pearson Education.


### Reference Books


1. Dietel & Dietel: Java How To Program, Pearson Education.

2. Bruce Eckel: Thinking in Java, Pearson Education, 2006
3. Balagurusamy: Programming with Java: A Primer (5th Ed.), 2014, Tata McGraw Hill.
4. Antonio Goncalves: Beginning Java™ EE 6 Platform with GlassFish 3 From Novice to Professional Apress Publication.

**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Java White Paper Buzzwords, History of Java, Common Misconceptions, Choosing a Development Environment: Command-Line Tools, Running a Graphical Application. |
| **Week 2:** | Building and Running Applets; A Simple Java Program, Comments, Data Types, Variables, Operators, Input and Output, Control Flow, Big Numbers, Arrays. |
| **Week 3:** | Introduction to OOP, Predefined Classes, User Defined Classes, Static Fields and Methods, Method Parameters, Object Construction. |
| **Week 4:** | Packages, Class Path, Documentation Comments, Class Design; Inheritance: Super-classes and Subclasses, Types of Inheritance, Polymorphism. |
| **Week 5:** | Abstract class, Object: The Cosmic Super class, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration Classes, Reflection, Inheritance Guidelines, Interfaces. |
| **Week 6:** | String Handling APIs: String, Immutable String, Methods of String Class, StringBuffer, StringBuilder, StringTokenizer. |
| **Week 7:** | Exceptions: Dealing with Errors, Catching Exceptions, Guidelines for Using Exceptions, Assertions, Logging. |
| **Week 8:** | Generic Programming: Definition, Generic Methods, Bounds for Type Variables, Generic Code and VM, Restrictions and Limitations, Inheritance Rules for Generic Types, Reflection and Generics. |
| **Week 9:** | Collection Interfaces, Concrete Collections, The Collections Framework, Algorithms, Legacy Collections. |
| **Week 10:** | Multithreading: Threads, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues. |
| **Week 11:** | Thread-Safe Collections, Callable and Futures, Executors, Synchronizers. |
| **Week 12:** | Introduction to Swing, Creating a Frame, Positioning a Frame, Displaying |

Information in a Component, Displaying Images, Event Handling, Basics of Event Handling, Actions, Mouse Events, The AWT Event Hierarchy

**Week 13:** JDBC: Basic JDBC Programming Concepts, JDBC Drivers, Statements, Executing Queries, Result Sets.

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Database Management System with Oracle Based Programming

| | | |
|---|---|---|
| Course Code | : | **CSCC32 / CBSE32** |
| LTP | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12 |
| Course Status | : | Core course / Choice Based Skill Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course shall include an extensive coverage of conceptual and practical aspects of databases. It is an applications course, and will focus on problems in the field of databases. Students are not expected to have any prior knowledge of databases, but they must have good programming skills and a grasp of basic theoretical knowledge. This course is also designed to introduce programming in databases using PL/SQ.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Use various models for designing databases.
2. Understand the concepts for different databases.
3. Learn normalization techniques.
4. Understand the concepts of transaction management

## UNITWISE SYLLABUS

1 **Database:** Data, Database and Database Management System (DBMS); Database vs. Traditional File System Approach; Three Schema Architecture of DBMS and Data Independence; Categories of Database Management Systems: Hierarchical, Network and Relational Database Systems.

2 **Database Models**: Introduction, Categories of Database Models: High-level or Conceptual Data Models, Representational or Implementation Data Models, Low-level or Physical Data Models, Object Data Models. Entity relationship (ER) Model: Basic Concepts and their representations – Entity, Entity Type and Entity Set, Attributes and Keys, Relationships, Relationship Types, and Structural Constraints, Weak Entity, Naming Conventions & Design Issues in ER Model. ER and EER Diagrams.

3 **Relational Database Model:** Structure of Relational Model; Domains, Attributes, Tuples, and Relations; Characteristics of Relations; Relational Constraints – Domain Constraints, Key Constraints, Entity Integrity, and Referential Integrity Constraints; Relational Database Schema; Relational Algebra Operations – Select, Project, Rename, Union, Intersection, Set Difference, Join, and Division Operations; Aggregate Functions and Groupings.

4 **Structured Query Language (SQL):** Schema, Table and Domain Creation; Schema and Table Deletion; Table Modification; Insert, Delete, and Update Statements; SELECT-FROM-WHERE Structure; Renaming Attributes; Nested Queries and Set Comparisons; EXISTS and UNIQUE Functions; Aggregate Functions; Creating and Updating Views. Introduction to PL/SQL.

5 **Functional Dependencies and Normalization:** Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference Rules for Functional Dependencies; Normalization using Functional Dependencies – First Normal Form (INF), Second Normal Form (2NF), Third Normal Form (3NF), and Boyce-Codd Normal Form (BCNF); Multi-Valued Dependencies and Fourth Normal Form (4NF); Join Dependencies and Fifth Normal Form (5NF); Relation Decomposition and Insufficiency of Normal Forms; Dependency Preserving and Lossless Join Decompositions; Null Values and Dangling Tuples.

**Transaction Management and Concurrency Control:** Transaction Concept; Transaction State; Concurrent Executions; Serializability and Recoverability; Testing for Serializability. Concurrency Control – Lock-Based Protocols and Timestamp-Based Protocols.

**Text Books**


1 R. Elmasri and S. B. Navathe: Fundamentals of Database Systems, 5th Ed., Pearson Education.

2 A. Silberschatz, H. F. Korth and S. Sudarshan: Database System Concepts, 4th Ed., McGRAW-HILL

3. Date C J, An Introduction to Database Systems, Addision Wesley

4. Majumdar & Bhattacharya, "Database Management System", TMH


**Reference Books**


1 K. Loney and G. Koch: ORACLE 9i – The Complete Reference, Tata McGRAW-HILL.

2 Leon & Leon,"Database Management Systems", Vikas Publishing House

3. Bipin C. Desai, " An Introduction to Database Systems", Gagotia Publications

4. P.K. Yadav: Database Management System, S.K.Katria and Sons Publication.


## Outline

**Week** **Topics**

**Week 1:** **Database:** Data, Database and Database Management System (DBMS); Database vs. Traditional File System Approach.

**Week 2:** Three Schema Architecture of DBMS and Data Independence; Categories of Database Management Systems: Hierarchical, Network and Relational Database Systems.

**Week 3:** **Database Models**: Introduction, Categories of Database Models: High-level or Conceptual Data Models

**Week 4:** Representational or Implementation Data Models, Low-level or Physical Data Models, Object Data Models.

**Week 5:** Entity relationship (ER) Model: Basic Concepts and their representations – Entity, Entity Type and Entity Set, Attributes and Keys, Relationships, Relationship Types, and Structural Constraints, Weak Entity, Naming Conventions & Design Issues in ER Model. ER and EER Diagrams

**Week 6:** **Relational Database Model:** Structure of Relational Model; Domains, Attributes, Tuples, and Relations; Characteristics of Relations; Relational Constraints – Domain Constraints, Key Constraints

**Week 7:** Entity Integrity, and Referential Integrity Constraints; Relational Database Schema; Relational Algebra Operations – Select, Project, Rename, Union, Intersection, Set Difference, Join, and Division Operations; Aggregate Functions and Groupings.

**Week 8:** **Structured Query Language (SQL):** Schema, Table and Domain Creation; Schema

and Table Deletion; Table Modification; Insert, Delete, and Update Statements; SELECT-FROM-WHERE Structure; Renaming Attributes

**Week 9:** Nested Queries and Set Comparisons; EXISTS and UNIQUE Functions; Aggregate Functions; Creating and Updating Views. Introduction to PL/SQL

**Week 10:** **Functional Dependencies and Normalization:** Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference Rules for Functional Dependencies; Normalization using Functional Dependencies

**Week 11:** First Normal Form (INF), Second Normal Form (2NF), Third Normal Form (3NF), and Boyce-Codd Normal Form (BCNF);

**Week 12:** Multi-Valued Dependencies and Fourth Normal Form (4NF); Join Dependencies and Fifth Normal Form (5NF); Relation Decomposition and Insufficiency of Normal Forms; Dependency Preserving and Lossless Join Decompositions; Null Values and Dangling Tuples.

**Week 13:** **Transaction Management and Concurrency Control:** Transaction Concept; Transaction State; Concurrent Executions; Serializability and Recoverability; Testing for Serializability. Concurrency Control – Lock-Based Protocols and Timestamp-Based Protocols; Deadlock Handling – Deadlock Preventions, Deadlock Detection and Recovery.

**Week 14:** Revision

### c) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Software Engineering with Minor Project (Definition & Design Phase)

| | | |
|---|---|---|
| Course Code | : | **CSCC33** |
| LTP | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11,12,23,25 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course will teach the students to effectively analyze and design a software system. It will focus on the application of a systematic, quantifiable approach for development, operation and maintenance of good quality software. This course aims at using sound engineering principles in order to obtain cost-effective software hat is reliable and works efficiently on real machines.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Use various tools for effective generation of the Software Requirements Specification document.
2. Use effective measurement tools to project cost estimates for a new project using effective software project management techniques.
3. Use tools for generation of a good design for the software being developed.
4. Use software metrics for effective estimations.
5. Study maintenance fundamentals and its measurement.
6. Communicate scientific knowledge at different levels of abstraction.

## UNITWISE SYLLABUS

**1. Software Engineering and Process:** Software and its Components; Evolving Role of software; Software Characteristics, Software: A Crisis on the Horizon , Legacy Software and Software Myths; Software Engineering – A Layered Technology.

**2. Software Process Models:** The Software Process, Software Process Models The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Software Process Models, Component-Based Development, The Formal Methods Model, Process Technology, Product and process.

3. **Design Concepts And Principles**: Software Design and Software Engineering, The Design Process, Design Principles, Design Concepts, Effective Modular Design, Design Heuristics for Effective Modularity, The Design Model, Design Documentation, Software Architecture, Data Design, Architectural Styles, Analyzing Alternative Architectural Designs, Mapping Requirements into a Software Architecture, Transform Mapping, Transaction Mapping, User Interface Design, Task Analysis and Modeling, Interface Design Activities, Implementation Tools, Design Evaluation, Structured Programming.

**4. Project Management Concepts**: The Management Spectrum, The Product, The Process, Software Process And Project Metrics: Measures, Metrics, and Indicators, Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Integrating Metrics Within the Software Engineering Process, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Data Modeling, Functional Modeling and Information Flow, Behavioral Modeling, The Mechanics of Structured Analysis, Software Design and Software Engineering.

**5. Software Testing and Quality:** Software Testing Fundamentals, Test Case Design, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Testing for Specialized Environments, Architectures, and Applications, Strategic Approach to Software Testing, Unit Testing, Integration Testing, Validation Testing, System Testing, The Art of Debugging. Quality Concepts, Software Quality Assurance and Software Quality Control.

### Text Books


1. Pressman: Software Engineering, 6th Ed.; Mc Graw Hill Publications.
2. Sommerville: Software Engineering, 7th Edition, Pearson Education.
3. Rajib Mall: Fundamentals of Software Engineering, 3 rd Edition, PHI, 2009.
4. kelkar S.A.: Software Engineering, PHI 2007.

**Reference Books**


1. Pankaj Jalote: Software Engineering, 3d ed, Narosa Publishing.
2. M Mehrotra & S Varshney: Principles and practices of Software Engineering, Narosa Pub.
3. Aggarwal & Singh: Software Engineering, 3ed; New Age International Publishers


**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Software and its Components; Evolving Role of software; Software Characteristics, Software: A Crisis on the Horizon |
| **Week 2:** | Legacy Software and Software Myths; Software Engineering – A Layered Technology. |
| **Week 3:** | The Software Process, Software Process Models The Linear Sequential Model, The Prototyping Model, The RAD Model. |
| **Week 4:** | Software Design and Software Engineering, The Design Process, Design Principles, Design Concepts, Effective Modular Design, Design Heuristics for Effective Modularity. |
| **Week 5:** | The Design Model, Design Documentation, Software Architecture, Data Design, Architectural Styles, Analyzing Alternative Architectural Designs, Mapping Requirements into a Software Architecture. |
| **Week 6:** | Transform Mapping, Transaction Mapping, User Interface Design, Task Analysis and Modeling, Interface Design Activities, Implementation Tools, Design Evaluation, Structured Programming. |
| **Week 7:** | The Management Spectrum, The Product, The Process, Software Process And Project Metrics: Measures, Metrics, and Indicators. |
| **Week 8:** | Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Integrating Metrics within the Software Engineering Process. |
| **Week 9:** | Software Project Estimation Decomposition Techniques, Empirical Estimation Models, Data Modeling, Functional Modeling and Information Flow. |
| **Week 10:** | Behavioral Modeling, The Mechanics of Structured Analysis, Software Design and Software Engineering. |
| **Week 11:** | Software Testing Fundamentals, Test Case Design, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing. |
| **Week 12:** | Testing for Specialized Environments, Architectures, and Applications, Strategic Approach to Software Testing, Unit Testing, Integration Testing, Validation Testing, System Testing. |

**Week 13:** The Art of Debugging. Quality Concepts, Software Quality Assurance and Software Quality Control.

**Week 14:** Revision

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc. Seminar presentation using ppt / multimedia, if needed:

**General Guidelines:**

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Analysis and Design of Algorithm

| | | |
|---|---|---|
| Course Code | : | **CSCC34** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11, CBSE12, CSCC14, CSCC23 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Techniques for the design and analysis of efficient algorithms, emphasizing methods useful in practice. Topics include a number of problem solving techniques like divide and conquer, dynamic programming, greedy approach, back tracking, and branch and bound for solving different kinds of problems. It also covers the different category of algorithms based on its time and space complexity.

**b) Objectives:** Upon completion of this course, students will be able to do the following:

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major problem solving techniques.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

## UNITWISE SYLLABUS

1. **Introduction:** Algorithm and its Basic features; Importance of Developing Efficient Algorithms; Every-Case Time Complexity, Worst-Case Time Complexity, Average-Case Time Complexity, and Best-Case Time Complexity; Complexity Representation using Order Notations: Big-o (O), Theta (Θ), Big-Omega

(Ω), Small-o (o) and Small-Omega (ω) Notations; Properties of Complexity Notations; Limit Approach to Determine Order, Master Theorem. Divide and Conquer Approach: Introduction to Divide-and-Conquer Approach – Divide, Conquer, and Combine Steps; Design and Analysis of Binary Search (Recursive and Non-recursive), Merger Sort, Quicksort, and Strassen's Matrix Multiplication Algorithms.

2. **Dynamic Programming:** Introduction to Dynamic Programming; Difference Between Divide-and-Conquer and Dynamic Programming Approaches; Binomial Coefficient Finding using Dynamic Programming; Dynamic Programming and Optimization Problems: Chained Matrix Multiplication and Longest Common Subsequence Problems; Travelling Salesman Problem.
3. **Greedy Approach:** Introduction to Greedy Approach; Components of Greedy Approach: Selection Procedure, Feasibility Check, and Solution Check; Minimum Spanning Tree Generation: Prim's and Kruskal's Algorithms; Dijkstra's Algorithm for Single-Source Shortest Paths; Scheduling: Single Server and Multi-Server Scheduling, Scheduling with Deadlines; Huffman Code; The Knapsack Problem (Greedy Approach vs Dynamic Programming): 0-1 Knapsack and Fractional Knapsack Problems.
4. **Backtracking:** Introduction to Backtracking; Backtracking Technique: State Space Tree, Promising and Non-Promising Nodes, Pruned State Space Tree; Backtracking Algorithms for n-Queens, Sum-of-Subsets, Graph Coloring, and 0-1 Knapsack Problems.
5. **Branch-and-Bound Method and Intractable Problems:** Introduction to Branch-and-Bound Method; Solving 0-1 Knapsack Problem using Branch-and-Bound Method: Breadth-First Search with Branch-and-Bound Pruning, Best-First Search with Branch-and-Bound Pruning; Solving Traveling Salesman Problem using Branch-and-Bound Method. Intractable Problems: NP-hard and NP-complete problems, Some examples of NP hard and NP complete Randomized Algorithm with examples.

**Text Books**


1. Neapolitan and Naimipour: Foundations of Algorithms using C++ Pseudo Codes, JBL.
2. Rizvi and Aggarwal, "Algorithms: Analysis, Design and Implementation", Anamaya Publications, 2007, ISBN: 978-81-89927-03-5
3. Horowitz and Sahani, Fundamentals of Computer Algorithms, Galgotia Publications Pvt Ltd Delhi India.
4. Aho, "Design and Analysis of Computer Algorithms", Pearson Education.


**Reference Books**


1. Thomas H Cormen Leiserson "Introduction to Algorithms", PHI Learning Private Limited, Delhi India.

2. Sara Baase and Allen Van Gelder, “Computer Algoritms : Introduction to Design and Analysis”, Pearson Education
3. Jon Kleinberg and Eva Tardos “Algorithm Design”, Pearson Education
4. Rizvi and Sharma, “Design and Analysis of Algorithms”, Nandani Prakashan Pvt. Ltd., New Delhi
5. Brassard Bratley “Fundamental of Algorithms”, PHI Learning Private Limited, Delhi India.
6. M T Goodrich “Algorithms Design”, John Wiley
7. Robert Sedgewick and Philippe Flajolet, “An Introduction to the Analysis of Algorithms”, 2nd Edition, Addison-Wesley,2013
8. George T. Heineman, Gary Pollice and Stanley Selkow, “Algorithms in a Nutshell”, O'Reilly Media, 2008.

**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Algorithms, Analysis of Algorithms, Design of Algorithms, Complexity of Algorithms, Asymptotic Notations |
| **Week 2:** | Master Theorem. Divide and Conquer Approach, Binary Search, Merger Sort, Quicksort, and Strassen’s Matrix Multiplication Algorithms |
| **Week 3:** | Dynamic Programming; Difference Between Divide-and-Conquer and Dynamic Programming Approaches; Binomial Coefficient Finding using Dynamic Programming; Dynamic Programming and Optimization Problems |
| **Week 4:** | Chained Matrix Multiplication and Longest Common Subsequence Problems; Travelling Salesman Problem |
| **Week 5:** | Greedy Approach; Components of Greedy Approach: Selection Procedure, Feasibility Check, and Solution Check; Minimum Spanning Tree Generation: Prim’s and Kruskal’s Algorithms |
| **Week 6:** | Dijkstra’s Algorithm for Single-Source Shortest Paths |
| **Week 7:** | Scheduling: Single Server and Multi-Server Scheduling, Scheduling with Deadlines; Huffman Code; The Knapsack Problem 0-1 Knapsack and Fractional Knapsack Problems |
| **Week 8:** | Backtracking Technique: State Space Tree, Promising and Non-Promising Nodes, Pruned State Space Tree; Backtracking Algorithms |

| | |
|---|---|
| | for n-Queens |
| **Week 9:** | Backtracking solution of Sum-of-Subsets, Graph Coloring, and 0-1 Knapsack Problems |
| **Week 10:** | Branch-and-Bound Method; Solving 0-1 Knapsack Problem using Branch-and-Bound Method |
| **Week 11 :** | Breadth-First Search with Branch-and-Bound Pruning, Best-First Search with Branch-and-Bound Pruning; Solving Traveling Salesman Problem using Branch-and-Bound Method. |
| **Week 12:** | Intractable Problems: NP-hard and NP-complete problems, Some examples of NP hard and NP complete. |
| **Week 13:** | Latest Advances in Algorithm, Best Seven Algorithm of Century |
| **Week 14:** | Revision |

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Computer Network and System Administration

| | | |
|---|---|---|
| Course Code | : | **CSCC35** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC11, CSCC12, CSCC21 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course will provide the knowledge and hands-on skills necessary to manage a Local Area Network and its resources. Topics covered include directory services, server management, file and print services, and user/client administration in a heterogeneous operating system environment. Students will setup and manage a fully functioning computer network of systems. Hands-on active learning required. During the courses, students will:

1. Use multiple computer system platforms, and understand the advantages of each.
2. Install and administer network services.
3. Protect and secure users' information on computer systems.
4. Use the command line interface for system administration.
5. Demonstrate strategies for planning/designing systems.
6. Install and manage disks and file systems.
7. Enable above learning outcomes in Windows and Linux environments.

**b) Objectives:** The educational objectives of the Computer Network and System Administration Program are to produce graduates who are able to:

1. Explain the importance of data communications and the Internet in supporting business communications and daily activities.

2. Explain how communication works in data networks and the Internet.
3. Recognize the different internetworking devices and their functions.
4. Explain the role of protocols in networking.
5. Analyze the services and features of the various layers of data networks.
6. Design, calculate, and apply subnet masks and addresses to fulfill networking requirements.
7. Undertake basic System Administration activities on popular operating system like Windows/Linux.

## UNITWISE SYLLABUS

1. **Introduction Concepts:** Goals and Applications of Networks, Network structure and architecture, The OSI reference model, services, Network Topology Design - Delay Analysis, Back Bone Design, Local Access Network Design, Physical Layer Transmission Media, Switching methods, ISDN, Terminal Handling.
2. **Medium Access sub layer:** Medium Access sub layer - Channel Allocations, LAN protocols - ALOHA protocols - Overview of IEEE standards - FDDI. Data Link Layer - Elementary Data Link Protocols, Sliding Window Protocols, Error Handling.
3. **Network Layer:** Network Layer - Point - to Pont Networks, routing, Congestion control Internetworking - TCP / IP, IP packet, IP address, IPv6.
4. **Transport Layer and Application Layer:** Transport Layer - Design issues, connection management, session Layer- Design issues, remote procedure call. Presentation Layer-Design issues, Data compression techniques, cryptography - TCP - Window Management.

   Application Layer: File Transfer, Access and Management, Electronic mail, Virtual Terminals, Other applications. Example Networks - Internet and Public Networks.
5. **Introduction to System Administration**: Concepts of users and groups, Access assignment and control, Group policies, File System Management, Disk quota for users, System logs, Backup/Restore operations, Remote access, setting up of firewalls, Firewll logs, Configuring network services such as DNS,DHCP,HTTP etc.

### Text Books


1. Computer Networks, Andrew S. Tanenbaum, PHI.
2. Introduction to Computer Networks, S.A.M. Rizvi, V.K. Sharma, Narosa.
3. Data and Computer Communications, William Stallings, PHI
4. Data Communications and Networking, Behrouz A. Forouzan, TMH.

**Reference Books**


1. Wireless Communications; Principles and Practice, Theodore E. Rappaport, 2nd Edition, Pearson Education .
2. Satellite Communications, Timothy Pratt, Charles Bostian and Jeremy Allnutt, 2nd Edition, Wiley
3. Internetworking with TCP/IP; Principles, Protocols and Architecture, Douglas E. Commer, 3rd Edition, Prentice Hall of India
4. Data Compression; The Complete Reference, David Solomon, 3rd Edition, Springer
5. Network Security; Private Communication in a Public World, CharlieKaufman, Radia Perlmon and Mike Speciner, 2nd Edition, Prentice Hall of India.
6. Windows 2003 and 2008 Server By BPB Publication
7. Windows XP Professional and Windows 7 Edition By BPB Publication
8. Red Hat Linux By BPB & SYBEX publication
9. Linux Bible By BPB & SYBEX publication


**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Goals and Applications of Networks, Network structure and architecture, The OSI reference model |
| **Week 2:** | Network Topology Design - Delay Analysis, Back Bone Design, Local Access Network Design |
| **Week 3:** | Physical Layer Transmission Media, Switching methods, ISDN, Terminal Handling |
| **Week 4:** | Medium Access sub layer: Medium Access sub layer - Channel Allocations |
| **Week 5:** | LAN protocols - ALOHA protocols - Overview of IEEE standards - FDDI |
| **Week 6:** | Data Link Layer - Elementary Data Link Protocols, Sliding Window protocols, Error Handling |
| **Week 7:** | Network Layer - Point - to Pont Networks, routing, Congestion control |
| **Week 8:** | Internetworking -TCP / IP, IP packet, IP address, IPv6. |
| **Week 9:** | Transport Layer - Design issues, connection management, session Layer- Design issues, remote procedure call. |
| **Week 10:** | Presentation Layer-Design issues, Data compression techniques, cryptography - TCP - Window Management. |

| | |
|---|---|
| **Week 11 :** | Application Layer: File Transfer, Access and Management, Electronic mail, Virtual Terminals, Other application. Example Networks - Internet and Public Networks. |
| **Week 12:** | Concepts of users and groups, Access assignment and control, Group policies, File System Management, Disk quota for users, System logs, Backup/Restore operations, Remote access. |
| **Week 13:** | Setting up of firewalls, Firewll logs, Configuring network services such as DNS,DHCP,HTTP etc. |
| **Week 14:** | Revision |

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Scientific and Statistical Techniques using FORTRAN/ R

| | | |
|---|---|---|
| Course Code | : | **CSCC36** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC14 |
| Course Status | : | Core Course |
| Instructor's Name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Numerical Analysis is concerned with mathematical derivation and analysis of methods to obtain the numerical solution of mathematical problems. Same time a number of statistical techniques may be used to get the statistical solution of a given problem. In research a hypothesis is derived and its validity is tested based on given data set. Therefore, this course is designed in such a way to fulfill these goals. It covers some basic techniques to get the numerical solutions and a number of statistical methods for hypothesis testing.

**b) Objectives:** Upon completion of this course, students will be able to do the following:

1. To get the roots of a complex algebraic and transcendental equations.
2. To solve system of simultaneous equations.
3. To get the numerical solutions of derivatives and integrations.
4. To define the hypothesis and test them.
5. To get the knowledge of probability distributions.

## UNITWISE SYLLABUS

1. **Solution of Equations and System of Simultaneous Equations:** Solution of Algebraic and Transcendental Equations using Bisection, Regula False, and Newton Raphson Methods, Solutions of Linear Systems using Matrix Inverse, Gauss Elimination, Gauss Seidel, and Jacobi Methods.
2. **Interpolation, Numerical Differentiation and Integration, and Differential Equations:** Interpolation using Lagrange, and Newton's methods, Extrapolation, Least Square Fitting, Numerical Differentiation, Numerical Integration using Trapezoidal, and Simpson's Rules, Numerical Solution of Ordinary Differential Equations using Euler's and Range-Kutta Methods.
3. **Statistics:** Population, Sample, Sample Collection Methods, Data Representations and Classification, Central Tendency and Dispersion: Mean, Geometric Mean, Harmonic Mean, Median and Mode, Quartiles and Percentiles, Measures of Dispersion: Range, Variance, Standard Deviation, and Coefficient of Variation.
4. **Probability, Correlation and Regression:** Sample Space, Events, Equally Likely Events, Probability, Independent Events, Addition and Multiplication Rules, Conditional Probability, Probability Distributions – Normal, Binomial, and Poison Distributions; Correlation using Karl Pearson and Spearman Method, Regression Analysis.
5. **Hypothesis**: Hypotheses, Hypothesis Testing, t-Test, Chi-Square Test, Analysis of Variance (ANOVA), One and Two Way ANOVA, F-Test.

### Text Books


1. S.S. Sastry, Introductory Methods of Numerical Analysis, PHI.
2. Gupta and Kapoor, Fundamentals of Mathematical Statistics a Modern Approach, S.Chand.
3. Seymour Lipschutz (Author), John J. Schiller: Introduction to Probability and Statistics, Mc Graw Hill.


### Reference Books


1. Phillips and Taylor, Theory and Applications of Numerical Analysis, Elsevier
2. Balagurusamy, Numerical Methods, TMH.
3. Lindsey, Introduction to Applied Statistics: A Modelling Approach, Oxford University Press

## c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Solution of Algebraic and Transcendental Equations using Bisection, Regula False, and Newton Raphson Methods |
| **Week 2:** | Solutions of Linear Systems using Matrix Inverse, Gauss Elimination |
| **Week 3:** | Solutions of Linear Systems using Gauss Seidel, and Jacobi Methods |
| **Week 4:** | Interpolation using Lagrange, and Newton's methods |
| **Week 5:** | Extrapolation, Least Square Fitting |
| **Week 6:** | Numerical Differentiation, Numerical Integration using Trapezoidal, and Simpson's Rules |
| **Week 7:** | Numerical Solution of Ordinary Differential Equations using Euler's and Range-Kutta Methods |
| **Week 8:** | Population, Sample, Sample Collection Methods, Data Representations and Classification |
| **Week 9:** | Central Tendency and Dispersion: Mean, Geometric Mean, Harmonic Mean, Median and Mode, Quartiles and Percentiles, Measures of Dispersion: Range, Variance, Standard Deviation, and Coefficient of Variation |
| **Week 10:** | Sample Space, Events, Equally Likely Events, Probability, Independent Events, Addition and Multiplication Rules, Conditional Probability |
| **Week 11 :** | Probability Distributions – Normal, Binomial, and Poison Distributions |
| **Week 12:** | Correlation using Karl Pearson and Spearman Method, Regression Analysis, Hypotheses, Hypothesis Testing, t-Test |
| **Week 13:** | Chi-Square Test, Analysis of Variance (ANOVA), One and Two Way ANOVA, F-Test |
| **Week 14:** | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Web Based Development using J2EE

| | | |
|---|---|---|
| Course Code | : | **CSCC41** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11, CBSE12, CBSE22, CBCS31 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Due to advancement in World Wide Web and Internet technologies, there have been a paradigm shift from desktop-based application to web-based server side application development. Hence, there is a need of web-based enterprise application developers in industry. There are many server side programming languages including Java 2 Enterprise Edition (J2EE), ASP.NET and PHP to name a few. The objective of this course is to train the students in the various aspects of Java – J2EE programming technologies with focus on ensuring understanding of the core concepts and ability to apply them to real world enterprise application development.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Fundamentals of Programming with World Wide Web.
2. Write Web-based applications using Java Servlets.
3. Write Web-based applications using Java Server Pages (JSPs).
4. Understand various concepts such as cookies, session management, database connectivity using Java Database Connectivity (JDBC).
5. Understand Web Servers and Servlet and JSP container, and how to deploy web-applications on them.

## UNITWISE SYLLABUS

1. **Internet Basics:** Overview of Internet, History, Web System Architecture – two-tier, three-tier and n-tier architecture, Internet vs. Intranet, Uniform Resource Locator, Protocol used in Internet – TCP/IP, HTTP, HTTPS, SMTP, POP, PPP, Services on the Internet - E-mail, Usenet, FTP, Search Engines, Web Browsers; Web Servers Introduction to J2EE, J2EE best practices.
2. **Java Database Connectivity, HTML and XML:** Java Database Connectivity – Different Types of Drivers, JDBC API's, Establishing a Connection, Statements & its Type, Result Set, Transactions Processing, Metadata. Basic Concept of HTML, Skeleton of a Web Page, Creating a Form, Overview of XML, XML Development Goal, Structure of XML Document, Parsing XML Documents – DOM and SAX.
3. **Java Servlet Programming:** Overview of CGI Programming, Benefits of Using Java Servlet, Servlet API Overview, Servlet Life Cycle, Servlet Implementation, Servlet Configuration, Servlet Exception, Deployment Descriptor, Servlet Deployment, Requests & Responses, Servlets & JDBC, Working with Cookies, Tracking Sessions, Context and Collaboration.
4. **Java Server Pages Basics and Architecture:** Introduction to Java Server Pages (JSP), JSP Tags, Directives, Scripting Elements, Standard Actions, Implicit Objects, Scope, JSP with Beans, JSP and Databases, Creating Custom JSP Tag Libraries using Nested Tags, Cookies User Sessions, Cookies and Session Objects.
5. **Introduction to Tomcat and EJB:** Web Container, Apache Tomcat Server, Architecture of Tomcat, Installing and Configuring Tomcat Server, Deeper Look into Deployment Descriptors, The EJB Container, EJB Classes, EJB Interfaces, , Session Java Bean, Entity Java Bean and Message-Driven Bean, The JAR Utility, Understanding JAR, WAR and EAR File Formats.

### Text Books


1. Jim Keogh: J2EE : The Complete Reference, Tata McGraw Hill.
2. Wrox Press: Professional JSP J2EE 1.3 Edition, Shroff Publishers.
3. Patrick Niemeyer Denial Leuck: Learning Java, OReilly Publication

**Reference Books**


1. Robert W. Sebesta: Programming the World Wide Web, 4th Ed., Addison Wesley, 2007.
2. Dustine R. Callway: Inside Servlet, Pearson Education.
3. James Goodwill: Developing Java Servlets, 2nd Ed., Sams.


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Overview of Internet, History, Web System Architecture – two-tier, three-tier and n-tier architecture, Internet vs. Intranet, Uniform Resource Locator. |
| **Week 2:** | Protocol used in Internet – TCP/IP, HTTP, HTTPS, SMTP, POP, PPP, Services on the Internet - E-mail, Usenet, FTP, Search Engines, Web Browsers; Introduction to J2EE, J2EE best practices. |
| **Week 3:** | Java Database Connectivity – Different Types of Drivers, JDBC API's, Establishing a Connection, Statements & its Type, Result Set. |
| **Week 4:** | Transactions Processing, Metadata. Basic Concept of HTML, Skeleton of a Web Page, Creating a Form. |
| **Week 5:** | Overview of XML, XML Development Goal, Structure of XML Document, Parsing XML Documents – DOM and SAX. |
| **Week 6:** | Overview of CGI Programming, Benefits of Using Java Servlet, Servlet API Overview, Servlet Life Cycle, Servlet Implementation. |
| **Week 7:** | Servlet Configuration, Servlet Exception, Deployment Descriptor, Servlet Deployment, Requests & Responses. |
| **Week 8:** | Servlets & JDBC, Working with Cookies, Tracking Sessions, Context and Collaboration. |
| **Week 9:** | Introduction to JSP, JSP Tags, Directives, Scripting Elements, Standard Actions, Implicit Objects. |
| **Week 10:** | Scope, JSP with Beans, JSP and Database, Creating Custom JSP Tag Libraries using Nested Tags. |
| **Week 11:** | Cookies User Sessions, Cookies and Session Objects. |
| **Week 12:** | Web Container, Apache Tomcat Server, Architecture of Tomcat, Installing and Configuring Tomcat Server. |
| **Week 13:** | Deeper Look into Deployment Descriptors. The EJB Container, EJB |

Classes, EJB Interfaces, Session Java Bean, Entity Java Bean and Message-Driven Bean, The JAR Utility, Understanding JAR, WAR and EAR File Formats.

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Artificial Intelligence and Prolog Programming

| | | |
|---|---|---|
| Course Code | : | **CSCC42 / CSAE42** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12 |
| Course Status | : | Core course / Choice Based Ability Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course presents AI as a coherent body of ideas and methods to acquaint the student with the classic programs in the field and their underlying theory. This course will introduce and familiarize students with the field of Artificial Intelligence (AI). AI is one of the oldest disciplines in computer science. A primary goal of this course is to build intelligent entities/systems. This course is structured to give an overview of the area, as well as provide necessary depth to fundamental AI techniques and applications. Main emphasis will be on investigations as what it means to be intelligent and how to incorporate intelligence in problem solving, throughout the course. Attempt will be made to attain an understanding on the contributions AI has made to the field of computer science. By the end of the course, students are expected to possess a general and in-depth knowledge of the field of AI. Students will explore this through problem-solving paradigms, search, logic and theorem proving, prolog language and non-monotonic reasoning, and learning. It is an applications course, and will focus on problems in the field of AI and techniques and algorithms for solving those problems using PROLOG. Prolog encourages a different programming style to Java or ML and particular focus is placed on programming to solve real problems that are suited to this style.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Use various symbolic knowledge representations to specify domains and reasoning tasks of a situated software agent.
2. Use different logical systems for inference over formal domain representations, and trace how a particular inference algorithm works on a given problem specification.
3. Understand the conceptual and computational trade-offs between the expressiveness of different formal representations.
4. To recognize when AI techniques are necessary, to apply standard AI techniques to solve problems
5. To introduce programming in the Prolog language and use key logic-based techniques for solving different AI problems
6. To understand non-monotonic reasoning or reasoning under uncertainty.

## UNITWISE SYLLABUS

1. **AI history and applications**: Introduction to Artificial Intelligence, Foundations and History of Artificial Intelligence, Defining AI: Acting Humanly (Turing Test Approach), Thinking Humanly (Cognitive Modeling Approach), Thinking Rationally (laws of thought approach), Acting Rationally (Rational Agent Approach); Foundations of Artificial Intelligence; AI techniques, Expert Systems, Applications of Artificial Intelligence, Intelligent Agents, Structure of Intelligent Agents. Introduction to Computer vision, Natural Language Possessing, Machine learning, Soft Computing etc.
2. **Problem solving using Search** : Searching for solutions, Uninformed Search Strategies: Breadth-first Search, Depth-first Search, Depth-limited Search, Iterative Deepening depth-first search, Comparing uninformed search strategies; constraint satisfaction problems, **Heuristic Search Techniques:** Hill Climbing, Simulated Annealing, Best First Search: OR Graphs, Heuristic Functions, A* Algorithm, AND-OR Graphs, AO* Algorithm, **Adversarial Search:** Zero-sum perfect information Games, Optimal Decisions and Strategies in Games, Mini-max Algorithm, Alpha-beta Pruning, Imperfect Real-time decisions, Games that include chance, State of the art game programs.
3. **Knowledge Representation & Reasoning**: Propositional logic, Theory of first order logic, Inference in First order logic, Forward & Backward chaining, Resolution, Representations and mappings, Approaches to Knowledge Representation, Procedural versus Declarative Knowledge; Predicate Logic: Representing Simple facts, Instance and is-a relationships in Logic, Proposition versus Predicate Logic, Computable Functions and Predicates, Rules of Inferences and Resolution-refutation, Logic Programming and Horn Clauses; Weak Slot-and-Filler Structures: Semantic Nets, Frames; Introduction to Semantic Web and ontologies, Strong Slot-and-Filler Structures: Conceptual Dependency, Scripts
4. **AI Programming Language (PROLOG):** Introduction, How Prolog Works, Backtracking, CUT and FAIL operators, Built-in Goals, Negation, Lists, Syntax and built-in Functions, Basic list manipulation

functions in PROLOG, Predicates and Conditionals, Input, Output and Local Variables, Iteration and Recursion, recursive Lists processing, Search in Prolog: Breadth-first, depth-first, Best-first search for AI problem solving

5. **Probabilistic/Statistical Reasoning:** Probability and Bayes' Theorem, Certainty Factors and Rule-Based Systems, Bayesian Networks, Exact and approximate inference in Bayesian networks, Markov chains, Dempster-Shafer theory, Quantifying uncertainty, Intro to Fuzzy Logic; Non-monotonic Reasoning, Truth Maintenance Systems, probabilistic reasoning over time.

**Text Books**


1. Stuart Russel and Peter Norvig: Artificial Intelligence–A Modern Approach, 3rd Ed., 2012, Pearson Education, ISBN: 0-13-790395-2
2. Elaine Rich, Kevin Knight and B. Nair: Artificial Intelligence, 2009, Tata McGraw Hill, 3rd Ed, ISBN-10: 0-07-008770-9
3. Dan W. Patterson, "Artificial Intelligence and Expert Systems", Prentice Hall of India.
4. Ivan Bratko: PROLOG Programming, 3rd Ed., 2001, Pearson Education, ISBN: 81-7808-257-8


**Reference Books**


1. Michael Negnevitsky, Ärtificial Intelligence-A guide to intelligent systems", 2nd edition, Pearson Education
2. E. Charniak and D McDermott, "Introduction to Artificial Intelligence", Pearson Education
3. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence : a logical approach", Oxford University Press, 2004.
4. George F. Luger, "Artificial Intelligence: Structures and Strategies for complex problem solving", Fourth Edition, Pearson Education, 2002.
5. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers, 1998.


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction to Artificial Intelligence, Foundations and History of Artificial Intelligence, Applications of Artificial Intelligence |

**Week 2:** Problem solving as search, un-informed and informed search strategies

**Week 3:** Adversarial search strategies, Minimax with Alpha-beta cut-offs

**Week 4:** Knowledge representation methods and reasoning, Propositional logic, Theory of first order logic, Inference in First order logic, Logic Programming and Horn Clauses

**Week 5:** Semantic nets, frames, Conceptual-dependency, scripts, and semantic web

**Week 6:** Introduction to prolog, automatic Backtracking, CUT and FAIL operators, Built-in Goals, Negation, Lists

**Week 7:** Syntax and built-in Functions, Basic list manipulation functions in PROLOG, Predicates and Conditionals, Input, Output and Local Variables, Iteration and Recursion, recursive Lists processing.

**Week 8:** Search in Prolog: Breadth-first, depth-first, Best-first search for AI problem solving

**Week 9:** Probability and Bayes' Theorem, Certainty Factors and Rule-Based Systems, Bayesian Networks,

**Week 10:** Exact and approximate inference in Bayesian networks

**Week 11:** Markov Chains, Dempster-Shafer theory,

**Week 12:** Non-monotonic Reasoning, Truth Maintenance Systems

**Week 13:** Fuzzy Logic introduction, Probabilistic reasoning over time

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Software Project Management with Minor Project (Testing & Implementation)

| | | |
|---|---|---|
| Course Code | : | **CSCC43** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11-CBSE12/CBSE22/CSCC23/CBCS31 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course describes the key aspects of a software project. It begins with the job description of a software manager and then addresses those topics germane to successful software development management, including organizing the software development team; interfacing with other engineering organizations (systems engineering, quality assurance, configuration management, and test engineering); assessing development standards; selecting the best approach and tailoring the process model; estimating software cost and schedule; planning and documenting the plan; staffing the effort; managing software cost and schedule during development; risk engineering; and continuous process improvement. Personnel management topics, including performance evaluations, merit planning, skills building, and team building, are also covered. This course introduces software engineers aspiring to become technical team leaders or software project managers to the responsibilities of these roles. For those engineers who have advanced to a software development leadership position, this course offers formal training in software project management.

**b) Objectives:** Apply management skills and techniques to both commercial and government projects. Specifically, by the end of this course, student will be able to:

1. Differentiate between the skills and roles of functional and technical managers for software efforts and their relationship with other organizations.

2. Produce specific sections of the plan used to manage the software development and maintenance efforts.
3. Evaluate software project management practices within an organization and recommend practical improvements based upon your evaluation.
4. Apply schedule and cost techniques to determine a Basis for estimate.

## UNITWISE SYLLABUS

1. **Introduction:** Introduction and Software Project Planning Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document, Project Management Cycle, SPM Objectives, Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan, Structure of a Software Project Management Plan, Software project estimation, Estimation methods, Estimation models, Decision process.
2. **Project Organization and Scheduling Project Elements**: Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle, Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques, Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts.
3. **Project Monitoring and Control:** Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV), Cost Performance Index (CPI), Schedule Performance Index (SPI), Interpretation of Earned Value Indicators, Error Tracking, Software Reviews, Types of Review: Inspections, Deskchecks, Walkthroughs, Code Reviews, Pair Programming.
4. **Software Quality Assurance:** Testing Objectives, Testing Principles, Test Plans, Test Cases, Types of Testing, Levels of Testing, Test Strategies, Program Correctness, Program Verification & validation,, Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, The SEI Capability Maturity Model CMM), SQA Activities, Formal SQA Approaches: Proof of correctness, Statistical quality assurance, Cleanroom process.
5. **Project Management and Project Management Tools:** Software Configuration Management: Software Configuration Items and tasks, Baselines, Plan for Change, Change Control, Change Requests Management, Version Control, Risk Management: Risks and risk types, Risk Breakdown Structure (RBS), Risk Management Process: Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis, Software Project Management Tools: CASE Tools, Planning and Scheduling Tools, MS-Project.

**Text Books**


1. Royce, Software Project Management, Pearson Education.
2. Rizvi and Aggarwal, Software Project Management: Principles & Practices, Khanna Book Publishing Company, New Delhi
3. Pankaj Jalote, Software Project Management in Practice, , Pearson
4. Roger S. Pressman ,Software Engineering: A practical Approach,, McGraw-Hill


**Reference Books**


1. M. Cotterell, Software Project Management, Tata McGraw-Hill Publication.
2. S. A. Kelkar, Software Project Management, PHI Publication.
3. Mantel et al, Project Management – Core text Book, Wiley
4. Nageswara Rao Pusuluri, Software Testing Concepts and Tools, , DreamTech
5. Kieron Conway, Software Project Management, Dreamtech Press


## c) Outline

| Week | Topics |
|---|---|
| **Week 1** | Introduction and Software Project Planning Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document, Project Management Cycle, SPM Objectives. |
| **Week 2** | Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan. |
| **Week 3** | Structure of a Software Project Management Plan, Software project estimation, Estimation methods, Estimation models, Decision process. |
| **Week 4** | Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle. |
| **Week 5** | Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques. |
| **Week 6** | Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts. |
| **Week 7** | Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV), Cost Performance Index (CPI). |
| **Week 8** | Schedule Performance Index (SPI), Interpretation of Earned Value Indicators, Error Tracking, Software Reviews, Types of Review: Inspections, Deskchecks, Walkthroughs, Code Reviews, Pair Programming. |

| | |
|---|---|
| **Week 9** | Testing Objectives, Testing Principles, Test Plans, Test Cases, Types of Testing, Levels of Testing, Test Strategies, Program Correctness, |
| **Week 10** | Program Verification & validation,, Concept of Software Quality, Software Quality Attributes. |
| **Week 11** | Software Quality Metrics and Indicators, The SEI Capability Maturity Model CMM), SQA Activities, Formal SQA Approaches: Proof of correctness, Statistical quality assurance, Cleanroom process. |
| **Week 12** | Software Configuration Management: Software Configuration Items and tasks, Baselines, Plan for Change, Change Control, Change Requests Management, Version Control. |
| **Week 13** | Risk Management Process: Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis, Software Project Management Tools: CASE Tools, Planning and Scheduling Tools, MS-Project. |
| **Week 14** | Revision |

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):** The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# BIG DATA Analytic and Cloud Computing

| | | |
|---|---|---|
| Course Code | : | **CSCC44** |
| L-T-P | : | 3-1-4 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11,CBSE12,CSCC13,CBCS21,CBSE22,CSCC25, CBCS31, CSCC33 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Big Data Analytics in particular, how data gleaned from social media sites may be used for substantive social science research. Emphasis will be placed on the limitations of current research and general difficulties in using data from various social media sites. Cloud computing has evolved as a very important computing model, which enables information, software, and other shared resources to be provisioned over the network as services in an on-demand manner. Topics may include distributed computing models and technologies, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), virtualization, security and privacy issues, performance and systems issues, capacity planning, disaster recovery, Cloud OS, federated clouds, challenges in implementing clouds, data centers, hypervisor CPU and memory management, cloud hosted applications, and other advanced and research topics in cloud computing.

**b) Objectives:** Upon completion of this course, students will be able to do the following:

1. Will provide exposure to scientific research in Big Data.
2. Will provide exposure to scientific research in cloud computing.
3. Analyze the real world problems.
4. Students will be exposed to the current practices in cloud computing.

5. Will be able to demonstrate the concept of Operating systems.

## UNITWISE SYLLABUS

1. **Big Data Processing Architectures:** Big Data Technologies, Data Driven Architecture, Information Management and Lifecycle, Big Data Analytics, Visualization and Data Scientist, Implementing The "Big Data" Data., Understanding Big Data Analysis with Machine Learning, The Evolution of Analytic Scalability, The Evolution of Analytic Processes, Creating a Culture of Innovation and Discovery, Think Bigger. Choices in Setting up R for Business Analytics, R Interfaces, Manipulating Data, Exploring Data, Building Regression Models, Forecasting and Time Series Models.
2. **Design Patterns and MapReduce:** Design Patterns, MapReduce History, MapReduce and Hadoop, Refresher, Hadoop Example: Word Count, Pig and Hive, Writing Hadoop Map Reduce Programs, Integrating R and Hadoop, Learning Data Analytics with R and Hadoop.
3. **Introduction To cloud:** Virtualization concepts, Types of Virtualization & its benefits, Introduction to Various Virtualization OS , Vmware , KVM etc, HA/DR using Virtualization , Moving VMs, SAN backend concepts, Cloud Fundamentals, Cloud Building Blocks, Understanding Public & Private cloud environments, Cloud as IaaS, PaaS , SaaS Private Cloud Environment , Basics of Private cloud infrastructure QRM cloud demo, Public Cloud Environment
4. **Understanding & exploring:** Amazon Web services, Managing and Creating Amazon EC2 instances, Managing and Creating Amazon EBS volumes, Tata Cloud details & demo, Managing Hybrid Cloud environment, Setting up your own Cloud, How to build private cloud using open source tools, Understanding various cloud plugins, Setting up your own cloud environment, Auto provisioning, Custom images, Integrating tools like Nagios, Integration of Public and Private cloud, Future directions, Cloud Domain and scope of work, Cloud Computing Programming Introduction, Trends and market of cloud.
5. **Cloud Computing:** The Cloud, Cloud Application Architectures, The Value of Cloud Computing, Cloud Infrastructure Models, An Overview of Amazon Web Services, Amazon Cloud Computing: Amazon S3,Amazon EC2, Before The Move Into The Cloud: Know Your Software Licenses , The Shift to a Cloud Cost Model, Service Levels for Cloud Applications, Security, Disaster Recovery. Security: Data Security, Network Security, Host Security, Compromise Response, Disaster Recovery: Disaster Recovery Planning, Disasters in the Cloud, Disaster Management, Scaling A Cloud Infrastructure: Capacity Planning, Cloud Scale.

**Text Books**


1. Data Warehousing in the Age of Big Data by Krish Krishnan, Morgan Kaufmann, 2013.
2. Big Data Analytics with R and Hadoop by Vignesh Prajapati,2013.
3. Cloud Computing : A Practical Approach by Anthony T. Velte Toby J. Velte, Robert Elsenpeter, 2010 by The McGraw-Hill.
4. Cloud Computing: SaaS, PaaS, IaaS, Virtualization and more. by Dr. Kris Jamsa.


**Reference Books**


1. Principles of Big Data Preparing, Sharing, and Analyzing Complex Information, 1st Edition, by J Berman, published by Morgan Kaufmann,2013 .
2. "Big Data Analytics - From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph" By David Loshin, Morgan Kaufmann,2013.
3. Big Data Application Architecture Q&A: a Problem - Solution Approach Nitin Sawant, Himanshu Shah,2013.
4. Big Data Now: Current Perspectives from O'Reilly Radar By O'Reilly Radar Team, 2011.
5. Cloud Computing Bible by Barrie Sosinsky, Published by Wiley Publishing, 2011.
6. Cloud Computing for Dummies by Judith Hurwitz, Robin Bloor, Marcia Kaufman, and Dr. Fern Halper, Wiley Publishing, 2010.
7. Cloud Computing Theory And Practice Danc.Marinercus, Elsevier, 2013
8. Gary Lee, "Cloud Networking - Understanding Cloud-based Data Center Networks", Elsevier, 2014


## c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Big Data Processing Architectures, Big Data Technologies, Data Driven Architecture, Information Management and Lifecycle, Big Data Analytics. |
| **Week 2:** | Understanding Big Data Analysis with Machine Learning, The Evolution of Analytic Scalability, The Evolution of Analytic Processes |

| | |
|---|---|
| **Week 3:** | Creating a Culture of Innovation and Discovery, Think Bigger. Choices in Setting up R for Business Analytics. |
| **Week 4:** | Interfaces, Manipulating Data, Exploring Data, Building Regression Models, Forecasting and Time Series Models. |
| **Week 5:** | Design Patterns and MapReduce: Design Patterns, MapReduce History, MapReduce and Hadoop. |
| **Week 6:** | Refresher, Hadoop Example: Word Count, Pig and Hive, Writing Hadoop Map Reduce Programs. |
| **Week 7:** | Integrating R and Hadoop, Learning Data Analytics with R and Hadoop. |
| **Week 8:** | Introduction To cloud: Virtualization concepts, Types of Virtualization & its benefits, Introduction to Various Virtualization OS. |
| **Week 9:** | Vmware , KVM etc, HA/DR using Virtualization , Moving VMs, SAN backend concepts, |
| **Week 10:** | Cloud Fundamentals, Cloud Building Blocks, Understanding Public & Private cloud environments, Cloud as IaaS, PaaS , SaaS. |
| **Week 11 :** | Understanding & exploring Amazon Web services, Managing and Creating Amazon EC2 instances, Managing and Creating Amazon EBS volumes |
| **Week 12:** | Integration of Public and Private cloud, Future directions, Cloud Domain and scope of work, Cloud Computing Programming Introduction, Trends and market of cloud. |
| **Week 13:** | Cloud Computing : The Cloud, Cloud Application Architectures, The Value of Cloud Computing, Cloud Infrastructure Models, Security: Data Security, Network Security, Host Security, Compromise Response, Disaster Recovery: Disaster Recovery Planning |
| **Week 14:** | Revision |

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and
2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Leadership, Interpersonal and Group Dynamics

| | | |
|---|---|---|
| Course Code | : | CSCC45 |
| L-T-P | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC16 |
| Course Status | : | Core course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course will provide students with a foundational understanding of the knowledge and skills required to lead groups. This course on Leadership and Group Dynamics is specially designed to provide the opportunity to study the principles underlying the process of group action and interaction in social situations and in professional leadership and supervisory group situations. In the popular book Emotional Intelligence, Daniel Goleman (1994) states that interpersonal skills are far more important to success in organizations than previously believed: "At best, IQ contributes about 20 percent to the factors that determine life success, which leaves 80 percent to other forces." You might have heard stories of exceptionally talented engineers and managers whose careers are stymied because of their perceived inability to work with others. Significant evidences exist to suggest that your impact in organizations depends partly on your ability to get along with others. Application of group dynamics to counseling, personal growth and other psychologically-oriented groups is emphasized to improve emotional intelligence. Experiential activities are included to enhance learning (and personal understanding and awareness). This course is designed to provide prospective counselors, psychologists, and professionals in related professions with an intellectual and personal understanding of the principles of leadership, membership, and development in groups, including awareness of how one's personal and interpersonal psychological and social styles affect personal and professional behavior in groups.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Describe concepts and variables by which groups are defined.

2. Differentiate between different types of group work.
3. Describe several models of human development, treatment, and their application to groups.
4. Understand and describe major group dynamic factors, including leadership, membership, and group developmental stages.
5. Understand groups as systems or interactive teams.
6. Understand ethical guidelines for group counselors and related issues.
7. Describe factors involved in planning, selecting, and conducting groups.
8. Describe and understand the interrelationship among major group dynamic factors.
9. Discuss the differential application of various group dynamic principles to different types of groups and group situations.
10. Demonstrate appropriate involvement, support, and feedback as a small group member and understand the impact of one's personal and interpersonal style on others in groups.
11. Appreciate differences among persons and understand the dynamic tension between the need to work together and the need to prize individuality in groups.
12. Be aware of professional training standards for group leaders.

## UNITWISE SYLLABUS

1. **An Introduction to Leadership Processes:** Introduction Concept and meaning of leadership, Theories of Leadership: Traditional and Modern theories of leadership, Emotional Intelligence
2. **Managing Leadership in Modern Times:** Leadership Styles, Roles and Activities of Leadership, Leadership Skills
3. **Group Processes:** Groups, Group Development, Group Dynamics, Group Cohesiveness, Types of Teams, Creating Effective Teams, Conflict: Sources, Patterns, Levels and Resolution. Stress: Emergence and causes of Stress; Effects of stress and Intra-individual conflict, Coping Strategies, Negotiating Skills. Organizational Power and Politics
4. **Inter Group Processes:** Communication and Feedback: Role of communication, non-verbal communication, interpersonal communication, interactive communication. Decision Making: Decision-Making Process, types of decision making, Transactional Analysis (TA), Johari Window
5. **Organizational Processes:** Organization Culture Concept and Determinants, Organizational Effectiveness: Concept and Measurement, Organizational Change Access to Files, Command Line Arguments.

**Text Books**


1. Robbins, Stephen P. (2000) Organizational Behaviour, Prentice Hall, New Delhi
2. Aswathappa, K. (2000). Organisation Behaviour, Himalaya Publishing House, New Delhi.
3. Mc Shane, L.S., Von, Glinow A.M., Sharma, R.R. (2010). Organizational Behaviour. Tata McGraw-Hill, New Delhi
4. Prasad, L. M. (2000). Organisational Behaviour. Sultan Chand & Sons, New Delhi.


**Reference Books**


1. Hersey, Paul, Blanchard, K.H.and Johnson, Dewey E. (2011). Management of Organizational Behavior: Leading Human Resources. PHI Learning Pvt. Ltd., New Delhi
2. Newstorm, John W. and Keith Davis (1998). Organizational Behaviour: Human Behaviour at Work.Tata McGraw-Hill, New Delhi.
3. Corey, M.S., & Corey, G. (2006 or 2010). Groups: Process and Practice (7th or 8th edition). Belmont, CA: Thomson/Brooks Cole.
4. Cloud, H., & Townsend, J. Making small groups work: What every small group leader needs to know. Grand Rapids, MI: Zondervan.
5. Dimock, H.G. (1987) Groups: Leadership and group Development. San Diego, University Associates.
6. Napier, R.W., & Gershenfeld, M. K. (1987). Groups: Theory and Experience. Boston: Houghton, Mifflin Company.
7. Yalom, I.D., (1985).The Theory and Practice of Group Psychotherapy. New York: Basic Books.


**c) Outline**

| **Week** | **Topics** |
|---|---|
| **Week 1:** | Introduction Concept and meaning of leadership |
| **Week 2:** | Theories of Leadership: Traditional and Modern theories of leadership, |
| **Week 3:** | Emotional Intelligence, Leadership Styles |
| **Week 4:** | Roles and Activities of Leadership, Leadership Skills |
| **Week 5:** | Groups, Group Development, Group Dynamics, Group Cohesiveness |
| **Week 6:** | Types of Teams, Creating Effective Teams, |
| **Week 7:** | Conflict: Sources, Patterns, Levels and Resolution |
| **Week 8:** | Stress: Emergence and causes of Stress; Effects of stress and Intra-individual conflict . |
| **Week 9:** | Coping Strategies, Negotiating Skills. Organizational Power and Politics |
| **Week 10:** | Inter Group Processes, Communication and Feedback: Role of communication, non-verbal communication |

**Week 11:** Interpersonal communication, interactive communication. Decision Making: Decision-Making Process

**Week 12:** Types of decision making, Transactional Analysis (TA), Johari Window, Organization Culture Concept and Determinants

**Week 13:** Organizational Effectiveness, Concept and Measurement, Organizational Change Access to Files, Command Line Arguments

**Week 14:** Revision

### d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,

1. In-Sem Continuous Internal Assessments and Evaluations, and
2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.
3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.
4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.
5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Elective- I (CS & IT)

| | | |
|---|---|---|
| Course Code | : | CSCC46 |
| L-T-P | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11/CBSE12 |
| Course Status | : | Core course / Choice Based Skill Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**********

# Course Plan and Detail Syllabi for

# Digital Image Processing and GPU Programming

| | | |
|---|---|---|
| Course Code | : | **CSCC51** |
| L-T-P- | : | 3-1-2 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11/CBSE12/CSCC14/CBCS21 |
| Course Status | : | Core course / CBCS |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:**

This course is an introduction to image processing and image analysis techniques and concepts. Areas examined include: Imaging sensors and their principles; Image representation and storage, coding and compression techniques, lossy versus lossless; Techniques for noise reduction in images. Image enhancement including contrast manipulation, histogram equalization, edge highlighting; Filtering and transform techniques for image processing including two dimensional Fourier transforms, wavelets and convolution; Spatial transformations and image registration.

**Course content:**

**Overview**

Visual information plays an important role in many aspects of our life. Much of this information is represented by digital images. Digital image processing is ubiquitous, with applications including television, tomography, photography, printing, robot perception, and remote sensing. It emphasizes general principles of image processing, rather than specific applications. We expect to cover the following topics: image acquisition and display, color representations, image sampling and quantization, point operations, linear image filtering and correlation, image transforms and sub-band decompositions, contrast and color enhancement, image restoration, and image compression.

**b) Objectives**

At the end of the course, the students will be prepared to:

1. Demonstrate a broad understanding of the standard image processing issues and analysis techniques used in the commercial and scientific community.
2. Perform techniques to enhance of contrast and color, and thereby the visual perception, of contrast degraded imagery.
3. Remove noise and other imaging artefacts from real-world imagery using a variety of filtering techniques in both the spatial and frequency domain.
4. Demonstrate an understanding of spatial resampling, linear spatial transforms and registration techniques.
5. Employ such techniques to resample imagery and accurately register pairs of images.
6. Apply and understand image analysis techniques to imagery in order to detect structures such as edges, lines and corners.

## UNITWISE SYLLABUS

1 **Introduction to Image Processing:** Introduction to Digital Image Processing, Human visua system image acquisition, camera, sampling theory. Image resizing, color fundamentals, colo models histogram Processing, contrast and brightness adjustment, arithmetic/logic operation, Spatial filtering, contrast enhancement, edge sharpening.

2 **Image Transformation Techniques:** 2D Fourier transform, Frequency domain processing, ringing artifact, pixel operations, geometric processing. Image restoration, denoising, deblur discrete wavelet transform.

3 **Operations on Image:** Image segmentation, Edge detection, edge, linking, hough transform thresholding, Region based segmentation, watershed segmentation, motion-based segmentati

4 **Introduction of Information Theory:** Morphological image processing. Multi-resolution processing. Elements of information theory, Lossless and Lossy coding, Huffman coding, arithmetic coding, run-length coding.

5 **Introduction of Pattern Recognition and Classification Techniques :**Introduction, Desig principles of pattern recognition system Statistical Pattern recognition, Parameter estimation methods - Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Classification Techniques – Nearest Neighbor (NN) Rule,Bayes Classifier. Support Vector Machine (SVM), K – means clustering.

**Text Books**


1. Gonzalez, Woods, Eddins, Digital Image Processing Using MatLab, Prentice-Hall, 2003.
2. Stuart Russel and Peter Norvig, " Artificial Intelligence – A Modern Approach", Pearson Education.
3. K. Mehrotra, C. Mohan, and S. Ranka, Elements of Artificial Neural Networks, MIT Press, 1997
4. C. Looney, Pattern Recognition Using Neural Networks, Oxford University Press, 1997
5. R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification, 2nd edition, Wiley-Interscience.


**Reference Books**


1. Kenneth R. Castleman, Digital Image Processing, Pearson, 2006.

2.Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MA TLAB',Pearson Education, Inc., 2004.

3. D,E. Dudgeon and RM. Mersereau, , Multidimensional Digital Signal Processing', Prentice Hall Professional Technical Reference, 1990.

4. William K. Pratt, , Digital Image Processing' , John Wiley, New York, 2002

5. Milan Sonka et aI, 'IMAGE PROCESSING, ANALYSIS AND MACHINE VISION', Broo kes/Cole, VikasPublishing House, 2nd edition, 1999,


**c)**

| **Weeks:** | **Topics:** |
|---|---|
| **Week 1:** | Introduction to Digital Image Processing, Human visual system image acquisition, camera, sampling theory. |
| **Week 2**: | Image resizing, color fundamentals, color models histogram, Processing, contrast and brightness adjustment |
| **Week 3:** | Arithmetic/logic operation, Spatial filtering, contrast enhancement, edge sharpening. |
| **Week 4:** | 2D Fourier transform, Frequency domain processing, ringing artifact, pixel operations, geometric processing. |
| **Week5:** | Image restoration, denoising, deblurring. |
| **Week 6 and 7:** | Image segmentation, Edge detection, edge, linking, hough transform, thresholding, Region based segmentation, watershed segmentation, motion-based segmentation. |

| | |
|---|---|
| **Week 8:** | Morphological image processing. |
| **Week 9:** | Multi-resolution processing, discrete wavelet transform. |
| **Week 10:** | Elements of information theory, Lossless and Lossy coding, Huffman coding, arithmetic coding, run-length coding. |
| **Week 11:** | Introduction, Design principles of pattern recognition system Statistical Pattern recognition, |
| **Week 12:** | Parameter estimation methods - Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA). |
| **Week 13:** | Classification Techniques – Nearest Neighbor (NN) Rule,Bayes Classifier, Support Vector Machine (SVM), K – means clustering |
| **Week 14 :** | Revision |

**d) Application Pedagogy**

The assessment and evaluation process will be broadly classified with the following TWO components, viz. :

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department..

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

# Course Plan and Detail Syllabi for

# Data Mining and Data Warehousing

| | | |
|---|---|---|
| Course Code | : | **CSCC52 / CSAE52** |
| L-T-P | : | 3-1-2 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12 |
| Course Status | : | Core course / Choice Based Ability Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course teaches the data mining and data warehousing techniques which includes automated methods to analyze patterns and models for all kinds of data, with applications ranging from scientific discovery to business intelligence and analytics. This course provides a broad yet in-depth overview of data mining, integrating related concepts from machine learning and statistics. The main focus of the course includes exploratory data analysis, pattern mining, clustering, and classification. This course lays the basic foundations of these tasks and also covers cutting-edge topics such as classification methods, high-dimensional data analysis, and complex graphs and networks. Second half of the course proceeds with basic concepts of cluster analysis, and then study a set of typical clustering methodologies, algorithms, and applications. This includes partitioning methods such as k-means, hierarchical methods such as BIRCH, density-based methods such as DBSCAN alongwith discussion of different data mining and clustering tools. This course also aims to give students an introduction about text mining, graph mining and social network analysis.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. Use various exploratory data analysis, pattern mining, clustering, and classification for solving real life problems

2. To become familiar with different data mining tools such as WEKA, RapidMiner, KNIME, Orange, NLTK etc. (source: http://thenewstack.io/six-of-the-best-open-source-data-mining-tools/)

## UNITWISE SYLLABUS

1. **Data Mining**: Introduction, Data warehouses, Transactional databases, Advanced Data Information Systems and Applications, Data Mining Functionalities, Classification of data mining systems, data mining task primitives, Integration of data mining systems with a data warehouse systems, Data Preprocessing: Descriptive data summarization, Data cleaning, Data Integration and Transformation, Data Reduction, Data discretization and Concept hierarchy generation.
2. **Data Warehouse and OLAP technology:** Multidimensional data model, Data Warehouse architecture and Implementation: OLAP, ROLAP, MOLAP, HOLAP etc., Data Cubes, Indexing OLAP data, OLAP queries, Full Cube Computation, BUC, Star-cubing, Discovery-driven exploration of data cubes.
3. **Frequent Patterns, Associations and Classification:** Association Rules, Frequent Itemsets, Closed Itemsets, Apriori algorithm, Generating association rules from frequent itemsets, Mining Closed Frequent Itemsets, Correlation Analysis, Metarule guided mining of Association Rules, Constraint Pushing, Classification v/s Prediction methods, Classification by Decision Tree Induction, Bagging and Boosting.
4. **Data Mining techniques**: Rule-based Classification, Rule extraction from a Decision Tree, Support Vector Machines for linearly and non-linearly separable data, Classification by Association Rule Analysis, k-Nearest-Neighbor Classifiers, Case-based Reasoning, **Prediction:** Linear v/s Non-linear Regression, Accuracy and Error measures: Hold-out method, Cross-validation, Bootstrap, estimating confidence intervals, ROC curves
5. **Clustering:** Types of data in Cluster Analysis, Categorization of Clustering methods, Partitioning Methods: k-means, k-Medoids, CLARANS, Hierarchical Methods: BIRCH, ROCK, Desity-based Methods: DBSCAN, Grid-based Methods, Mining Time-series data, Introduction to Text Mining, Graph Mining, Social Network Analysis, and Web or Link Mining

### Text Books


1. Jiawei Han, Micheline Kamber (2012), Data Mining: Concepts and Techniques, Morgan Kaufmann series in data management systems, 3rd Edition, Publisher Elsevier, 2012, ISBN-9380931913, 9789380931913
2. Ian H. Witten, Eibe Frank, Mark A. Hall (2011), Data Mining: Practical Machine Learning Tools and Techniques, 3rd Edition, MK Elsevier

**Reference Books**


1. Mohammed J. Zaki, Wagner Meira, Jr (2014), Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, ISBN 978-0-521-76633-3 Hardback, 2014
2. Anand Rajaraman, Jeffrey David Ullman (2011), Mining of Massive Datasets, Cambridge University Press, 27-Oct-2011 .
3. D J Hand , Padhraic Smyth Heikki Mannila, Specifications of Principles of Data Mining, 1st Edition (Hardcover), MIT Press (MA), 2001.


### c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction, Data warehouses, Transactional databases, Advanced Data Information Systems and Applications |
| **Week 2:** | Data Preprocessing, descriptive data summarization, Data cleaning, Data Integration and Transformation, Data Reduction |
| **Week 3:** | Multidimensional data model, Data Warehouse architecture and Implementation: OLAP, ROLAP, MOLAP, HOLAP |
| **Week 4:** | Data Cubes, Indexing OLAP data, OLAP queries, Full Cube Computation, BUC, Star-cubing |
| **Week 5:** | Association Rules, Frequent Itemsets, Closed Itemsets, Apriori algorithm |
| **Week 6:** | Mining Closed Frequent Itemsets, Correlation Analysis, |
| **Week 7:** | Data mining techniques: Decision tree and SVM |
| **Week 8:** | Classification by Association Rule Analysis, |
| **Week 9:** | k-Nearest-Neighbor Classifiers, Case-based Reasoning |
| **Week 10:** | Prediction using Regression |
| **Week 11:** | Hold-out method, Cross-validation, Bootstrap, estimating confidence intervals, ROC curves |
| **Week 12:** | Types of data in Cluster Analysis, Categorization of Clustering methods, Partitioning Methods: k-means |
| **Week 13:** | Hierarchical Methods: BIRCH, ROCK, Desity-based Methods: DBSCAN and brief introduction to Text Mining, Graph Mining, Social Network Analysis, and Web Mining |
| **Week 14:** | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Machine Learning and Soft Computing

| | | |
|---|---|---|
| Course Code | : | **CSCC53 / CSAE53** |
| L-T-P | : | 3-1-2 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC/CBCS11 / CSCC/CBSE12/CSCC34 |
| Course Status | : | Core course / Choice Based Ability Enhancement |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** This course presents the issues and techniques involved in the creation of computer programs that engage in intelligent behavior. Machine learning (ML) is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. ML is defined as a field of study that gives computers the ability to learn without being explicitly programmed. The goal of this course is to present the key algorithms and theory that form the core of machine learning. This course aims to present ML concepts drawn from many fields, including statistics, artificial intelligence, biology, philosophy, information theory, cognitive science, and control theory. The following are among the topics that would be covered: Genetic Algorithm (GA), Artificial Neural Network (ANN) and Soft Computing (Fuzzy, Neuro-Fuzzy etc). Experiments based on ANN, Fuzzy Logic and GA will be done using MATLAB.

**b) Objectives:** Subject-specific skills: By the end of this course, the student must be able to:

1. To endow students with sufficient theoretical and practical knowledge before they begin their doctoral research.
2. To enable students to write programs that improve automatically with experience eg. Face recognition, hand-written character recognition, machine translation etc.

3. To give in-depth knowledge about artificial neural network and evolutionary computation.

## UNITWISE SYLLABUS

1. **Machine Learning and applications**: Machine Learning, Applications of ML, Intelligent Systems, Introduction to Computer vision, Natural Language Possessing, Soft Computing etc, Types of Learning: Supervised, Unsupervised, and Reinforcement Learning, Decision Tree learning, Bayesian Learning and Statistical natural language processing
2. **Connectionist Models/ANN:** Foundations for Connectionist Networks, Biological Inspiration; Different Architectures and Output Functions: Feed-forward, Feedback, Recurrent Networks, Step, Sigmoid and Sigmoid Function; Different Models: MacCulloch and Pitts Model, Hopfield Model and Memories, Boltzmann Machines and Energy Computations, Learning Problems and Issues in feed-forward model: Supervised learning, Perceptron Learning, Delta rule and Backpropagation Learning.
3. **Unsupervised Learning with ANN**: Competitive Learning, Hebbian Coincidence Learning, Attractor Networks, SOM, Adaptive Resonance Theory (ART): Architecture, classifications, Implementation and training, Introduction to Deep learning and Ensemble methods
4. **Genetic Algorithm and Applications:** Introduction to genetic algorithms (GA), encoding, fitness functions, genetic operators, reproduction, evolutionary strategies, Applications of GA in Data Mining, and other applications e.g. Travelling Salesman problem, differential evolution, co-evolution, multi-objective GA (MOGA), Neuro-Genetic hybrid algorithm; Swarm Intelligence: Introduction, Swarm Based versus Population based techniques, Particle Swarm Optimization, Ant Colony Optimization
5. **Fuzzy Logic:** Introduction to Fuzzy Systems – fuzzy sets: properties and operations: union, intersection, complement, s-norm, t-norm, alpha-cut; Fuzzy logic and fuzzy rules, Mamdani fuzzy rule inferencing mechanism, Fuzzy logic based Systems: e.g. Room Cooler; introduction to Neuro Fuzzy Systems: modeling fuzzy neuron, fuzzy neural network etc.

### Text Books


1. Tom M. Mitchell: Machine Learning, McGraw-Hill, International Edition 1997, ISBN 0-07-115467-1
2. Elaine Rich, Kevin Knight and B. Nair: Artificial Intelligence, 2009, Tata McGraw Hill, 3rd Ed, ISBN-10: 0-07-008770-9
3. S. Rakasekharan, GA Vijayalakshmi, Neural Networks, Fuzzy Logic and Genetic Algorithms, PHI.
4. Stuart Russel and Peter Norvig: Artificial Intelligence–A Modern Approach, 3rd Ed., 2012, Pearson Education, ISBN: 0-13-790395-2 (2nd edition downloadable)

**Reference Books**


1. Michael Negnevitsky, Ärtificial Intelligence-A guide to intelligent systems", 2nd edition, Pearson Education
2. A.P. Engelbrecht, Computational Intelligence: An Introduction, Wiley.
3. Jyh-Shing Roger Jang, Cuen Tsai Sun, EijiMizurani, Neuro Fuzzy and Soft Computing, A Computational Approach to Learning and Machine, PEA.
4. Padhy, N. P. (2005). *Artificial intelligence and intelligent systems*. Oxford University Press.
5. S.N. Sivanandam & S.N. Deepa, Principles of Soft Computing, Wiley Publications.
6. Deb, Kalyanmoy (1999). Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary computation*, *7*(3), 205-230.
7. B. Yegna Narayana: Artificial Neural Network, EEE, PHI, 2001, ISBN 81-203-1253-8


## c) Outline

| Week | Topics |
|---|---|
| **Week 1:** | Introduction to Machine Learning and Decision Tree Learning |
| **Week 2:** | Statistical natural language processing |
| **Week 3:** | Feed-back architecture and Hopfield Model |
| **Week 4:** | Feed-forward architecture and Backpropagatin learning |
| **Week 5:** | Unsupervised Learning and SOM |
| **Week 6:** | Demo of MATLAB toolboxes for ANN, GA and Fuzzy Logic |
| **Week 7:** | Adaptive Resonance Theory (ART) |
| **Week 8:** | Introduction to Deep Learning and Ensemble methods |
| **Week 9:** | Genetic algorithms and its applications |
| **Week 10:** | Multi-objective genetic algorithms |
| **Week 11:** | Introduction to Fuzzy Systems |
| **Week 12:** | Sugeno and Mamdani fuzzy inferencing and application |
| **Week 13:** | Neuro Fuzzy Systems, Designing and implementation system of a machine learning system |
| **Week 14:** | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and

2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**:

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Course Plan and Detail Syllabi for

# Pattern Matching using Python Programming

| | | |
|---|---|---|
| Course Code | : | **CSCC54** |
| L-T-P | : | 3-1-2 |
| Credit | : | 4 |
| Course Prerequisite | : | CSCC11, CSCC12, CSCC31 |
| Course Status | : | Core Course |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**a) Course Description:** Computer technology and applications have become increasingly more sophisticated over the past two decades, and so has the computer science curriculum. Today's students learn a bit of programming and problem–solving, and are then expected to move quickly into topics like software development, complexity analysis, and data structures that, twenty years ago, were relegated to advanced courses. In addition, the ascent of object-oriented programming as the dominant paradigm of problem solving has led instructors and textbook authors to bring powerful, industrial-strength programming languages such as C++ and Java into the introductory curriculum. As a result, instead of experiencing the rewards and excitement of solving problems with computers, beginning computer science students often become overwhelmed by the combined tasks of mastering advanced concepts as well as the syntax of a programming language.This course uses the Python programming language as a way of making the final year of computer science more manageable and attractive for students and instructors alike. Python has the following pedagogical benefits:

- Python has simple, conventional syntax. Python statements are very close to those of pseudocode algorithms, and Python expressions use the conventional notation found in algebra. Thus, students can spend less time learning the syntax of a programming language and more time learning to solve interesting problems like pattern matching, Database searching etc..
- Python has safe semantics. Any expression or statement whose meaning violates the definition of the language produces an error message.

- Python scales well. It is very easy to write simple programs in Python. Python also includes all of the advanced features of a modern programming language, such as support for data structures and object-oriented software development, for use when they become necessary.
- Python is highly interactive. Expressions and statements can be entered at an interpreter's prompts to allow the programmer to try out experimental code and receive immediate feedback. Longer code segments can then be composed and saved in script files to be loaded and run as modules or standalone applications.
- Python is general purpose. In today's context, this means that the language includes resources for contemporary applications, including media computing and networks.
- Python is free and is in widespread use in industry.

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

- Execute Python code in a variety of environments
- Use correct Python syntax in Python programs
- Use the correct Python control flow construct
- Write Python programs using various collection data types
- Write home grown Python functions
- Use many of the standard Python modules
- Trap various errors via the Python Exception Handling model
- Use the IO model in Python to read and write disk files
- Create their own classes and use existing Python classes
- Understand and use the Object Oriented paradigm in Python programs
- Use the Python Regular Expression capabilities for data verification and Pattern Matching.

**b) Objectives:** The educational objectives of the Pattern matching using Python Program are to produce graduates who are able to:

1. Combine some of the best features of mathematics, engineering, and natural science.
2. Understand the meaning of Problem solving by learning the ability to formulate problems, think creatively about solutions, and express a solution clearly and accurately.
3. Analysis and interpretation of Data.
4. Design and provide proper structure for the unstrutured data.
5. Reduce the data matching, searching and processing time

## UNITWISE SYLLABUS

**1. Introduction Concepts:** History, Features, Setting up path, Working with Python, Basic Syntax variable and Data Types,Operators, conditional statements,Looping statements,control statements. Input/output: Printing on Screen, Reading data from keyboard, Opening and closing file, Reading and writing files, I/O Functions.

**2. String Manipulation and Pattern Matching**: Accessing Strings, Basic Operations, String slices, Functions and Methods, Regular Expressions, Match function, Search function, Matching vs Searching, Modifiers, Patterns, Pattern Matching.

**3. Basic Data Structures and OOP concepts:** List, Accessing list, Working with lists, Operations, related Functions and Methods, Tuple, Accessing tuples, Working with tuples, Operations, related Functions and Methods, Dictionary, Working with dictionary, Accessing values in dictionaries, Working with dictionaries, Operations, related Functions and Methods. OOPs concepts, Class and object, Attributes, Inheritance, Overloading, Overriding, Data hiding.

**4. Functions and Modules:** Defining a function, Calling a function, Types of functions Function Arguments, Anonymous functions, Global and local variables, Introduction to Modules, Importing module, Math module, Random module, Packages.

**5. CGI and Database:** Introduction, Architecture, CGI environment variable, GET and POST methods Cookies, File upload, Database connectivity, Introduction, Connections, Executing queries, Transaction.

**Text Books**


1. Think Python- How to think like a Computer Scientist, By Allen B. Downey, Green Teen press, 2008.
2. Fundamental of Python - By Kenneth A. Lambert, Course Technology, 2010.
3. Alberto Apostolico, Zvi Galil: Pattern Matching algorithms, Oxford University Press.


**Reference Books**


1. Stuart Russel and Peter Norvig: Artificial Intelligence–A Modern Approach, 3rd Ed., 2012, Pearson Education.

2. Think Python, by Allen B. Downey, second edition, O'Reilly, Sebastopol, California.


**c) Outline**

| Week | Topics |
|---|---|
| **Week 1:** | Introduction |
| **Week 2:** | Basic Concepts of Python |
| **Week 3:** | String Manipulation- Basic Operations |
| **Week 4:** | Pattern Matching , Searching and Data processing |
| **Week 5:** | Basic Data Structures List, Tuple and Dictionary. |
| **Week 6:** | Functions- Inbuilt and User-Defined |
| **Week 7:** | Basic Operations and Functions related to basic Data structures |
| **Week 8:** | Basic OOPs Concepts in Python |
| **Week 9:** | Inheritance,Overloading, Overriding, Data hiding. |
| **Week 10:** | Modules- Inbuilt and User-Defined |
| **Week 11 :** | Important Packages of Python related to Data Processing |
| **Week 12:** | Database Connectivity, CGI Concepts |

| | |
|---|---|
| **Week 13:** | Dynamic Web Designing, Building a small real life system using overall Python features. |
| **Week 14:** | Revision |

## d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and
2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

**Various Modes of Evaluations (10%):**

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

**Quiz Schedule**:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

**Assignments, if any:**

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.

2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.

3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.

4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.

5. Grading System and Mark sheet will be issued as per JMI rules.

**********

# Elective- II (CS & IT)

| | | |
|---|---|---|
| Course Code | : | **CSEC55** |
| L-T-P | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11/CBSE12 |
| Course Status | : | |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**********

# Elective- III (Allied)

| | | |
|---|---|---|
| Course Code | : | **CSEC56** |
| L-T-P | : | 3-1-0 |
| Credit | : | 4 |
| Course Prerequisite | : | CBCS11/CBSE12 |
| Course Status | : | |
| Instructor's name | : | |
| Tel. No. | : | |
| Office Location | : | |
| Office Hours | : | |
| Class Location | : | |
| Class Time | : | |

**********

# Annexure – I

# PGDCA Curriculum

## 1. INTRODUCTION

Post Graduate Diploma in Computer Applications (PGDCA) is a 1-year (2-Semesters) length academic, regular programme of professional nature. It is primarily aimed to cater the rapidly increasing requirements of skilled Technical Assistance in modern IT-enabled system operations, developments and maintenance. Moreover, aspirants may take it as a foundation programme to lateral scheme entries to Master's programme in IT, Computer Science, etc. as well as an add-on qualification for professional growth.

## 2. PROGRAMME HIGHLIGHTS

| | | |
|---|---|---|
| **A.** | Programme Name | Post Graduate Diploma in Computer Applications (PGDCA) |
| **B.** | Nature | Regular (Credit-Based Semester System) |
| **C.** | Duration | 1 Year (2 Semesters) |
| **D.** | Programme Credits | 50 |
| **E.** | Theoretical Courses | 03 Courses (12 Credits) |
| **F.** | Lab-Oriented Courses | 04 Courses (08 Credits) |
| **G.** | Skill Development Courses | 06 Courses (24 Credits) |
| **E.** | Minor Project | 01 (06 Credits) |
| **F.** | OS Platforms | Windows, Unix, and Linux |
| **G.** | Programming Languages | C, C++, VB.NET, ASP & HTML |
| **H.** | Databases | Oracle |
| **I.** | Software Tools | MS Office & FLASH MX |

## 3. PROGRAMME STRUCTURE

In order to achieve the aims and objectives set forth at the outset, a programme structure under the semester-based credit system is prescribed as follows. The programme structure prescriptions include the titles of the theory/lab courses along with L-T-P and assigned credits. Further the respective detailed syllabi of theory courses are based largely on the latest available edition of the book prescribed first in the list, with minor

portions or special-topics-of-interest based on the respective reference books. Moreover, prescriptions of ten generic practical assignments are included as suggestive exercises, on which specific assignments for student(s) may be worked upon by the concerned teachers.

**PGDCA: PROGRAMME STRUCTURE**

| SEM | CODE | PAPER-TITLE | L-T-P | Marks | CR | SUMMARY |
|---|---|---|---|---|---|---|
| I | DCA11 | Computer Fundamentals | 3-1-0 | 100 | 4 | L:T:P = 15:5:10<br>Credit=24 |
| | DCA12 | Problem Solving and Programming in C | 3-1-0 | 100 | 4 | |
| | DCA13 | DBMS with Oracle based Programming | 3-1-0 | 100 | 4 | |
| | DCA14 | Multimedia Applications | 3-1-2 | 100 | 4 | |
| | DCA15 | Networking Technologies | 3-1-0 | 100 | 4 | |
| | DCA16 | Lab-I (Programming in C) | 0-0-4 | 100 | 2 | |
| | DCA17 | Lab-II (Oracle) | 0-0-4 | 100 | 2 | |
| II | DCA21 | Visual Programming | 3-1-0 | 100 | 4 | L:T:P = 12:5:18<br>Credit=26 |
| | DCA22 | Operating System and Shell Programming | 3-1-0 | 100 | 4 | |
| | DCA23 | Website Design and Management | 3-1-2 | 100 | 4 | |
| | DCA24 | Object Oriented Programming using C++ | 3-1-0 | 100 | 4 | |
| | DCA25 | Lab-III (VB.NET) | 0-0-4 | 100 | 2 | |
| | DCA26 | Lab-IV (Programming in C++) | 0-0-4 | 100 | 2 | |
| | DCA27 | Minor Project | 0-2-8 | 100 | 6 | |
| TOTAL CREDITS | | | | | | 50 |

# PGDCA DETAILED SYLLABI

| DCA11 | Computer Fundamentals | 3-1-0 | 4 |
|---|---|---|---|

1. **Computing Concepts:** Basic Computing Systems, Layers of a Computing System, History of Computing, History of Computing Software, Stored-Program Concept and von Neumann Architecture. Fetch-Execute Cycle, RAM and ROM, Types of RAM and ROM, Secondary and Tertiary Storage Devices, Cache Memory, Memory Hierarchy, Input-Output Devices, Touch Screens.
2. **Data Representation and Logic Gates:** Binary Values and Computers, Data and Computers, Analog and Digital Data; Binary Representation. Number Systems: Binary, Octal, Decimal, and Hexadecimal. Conversions of Data from one Number System to another Number System. Representation of Numeric Data – Negatives and Real Data Representation. Representing Texts - ASCII and Unicode Character Sets. Binary Arithmetics – Addition and Substraction of Numbers in Different Number Systems. Gates and Circuits: Computers and Electricity; Logic Gates – AND, OR, NOT, XOR, NAND and NOR Gates. Gate Processing; Gates with More Inputs; Constructing Gates; Transistors; Circuits – Combinatorial Circuits: Adders and Multiplexers. Circuit as Memory; Integrated Circuits; CPU Chips.
3. **Programming Languages:** Computer Operations; Levels of Abstraction; Machine Language; Assembly Language; Pseudo-Operations; Introduction to Interpreter and Compiler, Programming Language Paradigms, Procedural vs. Object-Oriented Paradigms. Boolean Expressions; Strong Typing; Input-Output Structures; Control Structures; Composite Data Types.
4. **Information Systems:** Data vs Information; Information Systems and Technologies; Spreadsheets; Spreadsheet Formulas; Circular References; Spreadsheet Analysis; Database Management Systems; The Relational Model; Relationships; Structured Query Language; Database Design; Information Security; Confidentiality, Integrity and Availability; Cryptography.
5. **Systems Analysis and Design:** Need for Systems Analysis and Design; Systems Analyst - Role, Expertise, Qualities, and Responsibilities; System Development Life Cycles, Information Requirements Gathering, Analyzing Requirements, Development and Documentation; Use Case Tools; Software Engineering, Reverse Engineering, and Reengineering; OOAD, Software Testing and Quality Assurance.

**BOOKS**


**1.** Dale & Lewis: Computer Science Illuminated, Narosa Publishing House,
**2.** Kedall & Kendall: Systems Analysis and Design, Prentice Hall India
**3.** Rajaraman: Fundamentals of Computers, Prentice Hall India
**4.** ITL Esl: Introduction to Computer Science, Pearson Education


+

| DCA12 | Problem Solving and Programming in C | 3-1-0 | 4 |
|---|---|---|---|

1. **Problem Solving Approach**: Introduction to Programs and Algorithms; Problem Solving Aspect (Algorithm Devising); Basic Features of an Algorithm; Algorithm Design Using Pseudo codes; Algorithm Implementation; Program Verification; Flowchart – Symbols and Conventions.
2. **Basic Concepts of C Programming Language:** Character Set; Keywords; Identifier, Constants, and Variables; Constant Types – Numeric and Character Constants; Data Types –Character, Integer and Floating Point; Signed, Unsigned, Short, and Long Integers; Data Declaration and Definition, Operator & Expression – Arithmetic, Relational, Logical, Increment, Decrement, Assignment, Conditional, and Bitwise Operators; Precedence & Associability of Operators; Managing Console I/O, Formatted I/O, Control Structures: Decision Making (Branching) Structures – If Statement, If-Else Statement, Nested If-Else Statement, Else-If Ladder, Switch Statement, Goto Statement; Looping Structures – While Statement, Do-While Statement, For Statement, Continue and Break Statements.

3. **Functions:** Library Functions; User-Defined Functions; Function Declaration (Prototype) and Function Definition; Function Arguments – Dummy, Actual and Formal Arguments; Local and Global Variables; Function Calls – Call by Value and Call by Reference; Returning Multiple Values from a Function, Recursion and Recursive Functions, Storage Class & Scope of Variables – Automatic Storage, Extern Storage, Static Storage, and Register Storage.
4. **Arrays, Strings & Pointers:** Single Dimensional Arrays; Accessing Array Elements; Initializing an Array; Multidimensional Arrays; Initializing Multidimensional Arrays; Memory Representation; Accessing Multidimensional Array Elements; Array of Characters; String Manipulation Functions; Introduction to Pointers; Pointer Variable Declarations and Initializations; Null Pointer; Constant Pointers; Void Pointer; Pointer Operators; Pointer Arithmetic; Application of Pointers; Dynamic Memory Allocations – malloc, calloc, realloc and free functions; Pointers and Strings.
5. **Structure, Union, Enumeration and Files:** Structure Declaration and Initialization; Accessing Structure Members, Structure Assignments; Array of Structures and Arrays within Structures, Nested Structures; Structure as Function Arguments; Structure Pointer; Unions; Difference between Structure and Union; Bit-Fields; Introduction to File; Text and Binary Files; Defining, Opening and Closing Files; I/O Operations on Files, Command Line Arguments.

**BOOKS**


**1.** T E. Balagruswamy: Programming in ANSI C, 3rd Ed., Tata McGraw Hill
**2.** R1 R. G. Dromey: How to Solve by Computer, 5th Ed., Pearson Education
**3.** R2 Deitel & Deitel: C – How to Program, 3rd Ed., Pearson Education


+

| DCA13 | DBMS with Oracle based Programming | 3-1-0 | 4 |
|---|---|---|---|

1. **Database and Database Models:** Data, Database and Database Management System (DBMS); Database vs. Traditional File System Approach; Three Schema Architecture of DBMS and Data Independence; Classification of Database Management Systems – Hierarchical, Network and Relational Database Systems; Centralized and Client-Server Architectures for DBMSs Database Languages and Interfaces; Database Users. **Database Models**: Introduction, Categories of Database Models: High-level or Conceptual Data Models, Representational or Implementation Data Models, Low-level or Physical Data Models, Object Data Models. Entity relationship (ER) Model: Basic Concepts and their representations – Entity, Entity Type and Entity Set, Attributes and Keys, Relationships, Relationship Types, and Structural Constraints, Weak Entity, Naming Conventions & Design Issues in ER Model. ER Diagrams.
2. **Relational Database Model:** Structure of Relational Model; Domains, Attributes, Tuples, and Relations; Characteristics of Relations; Relational Constraints – Domain Constraints, Key Constraints, Entity Integrity, and Referential Integrity Constraints; Relational Database Schema; Relational Algebra Operations – Select, Project, Rename, Union, Intersection, Set Difference, Join, and Division Operations; Aggregate Functions and Groupings.
3. **Functional Dependencies and Normalization:** Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference Rules for Functional Dependencies; Normalization using Functional Dependencies – First Normal Form (INF), Second Normal Form (2NF), Third Normal Form (3NF), and Boyce-Codd Normal Form (BCNF); Relation Decomposition and Insufficiency of Normal Forms; Dependency Preserving and Lossless Join Decompositions; Null Values and Dangling Tuples.
4. **Introduction to Oracle and Structured Query Language (SQL):** Introduction to Oracle; Features of Oracle; Form Design; Schema and Table Creation; Schema and Table Deletion; Table Modification; Insert, Delete, and Update Statements; SELECT-FROM-WHERE Structure; Renaming Attributes; Nested Queries and Set Comparisons; EXISTS and UNIQUE Functions; Aggregate Functions; Creating and Updating Views.

5. **PL/SQL:** Introduction to PL/SQL, Handling Data in PL/SQL Blocks; PL/SQL Processing; Programming Constructs, Procedures, Functions, Exception handling, PL/SQL Packages; Database Triggers; Oracle-Supplied Packages. Miscellaneous topics and supplements

**BOOKS**


**1.** Navathe: Fundamentals of Database Systems**,** Pearson Education
**2.** Ivan Bayross: SQL, Pl/SQL – The Programming Language of Oracle, BPB Publication
**3.** Rosenzweig and Silvestrova: Oracle PL/SQL by Example, Pearson Education


+

| DCA14 | Multimedia Applications | 3-1-2 | 4 |
|---|---|---|---|

1. **Multimedia Primer:** Basic Concepts, Multimedia Storage Devices, Multimedia Highway, Multimedia Applications; Stages in Multimedia Development; Multimedia Development Requirements, Multimedia Skills Development and Expertise Requirements.
2. **Multimedia Text, Sound, Images, and Video:** Text-Power and Meaning, Fonts and Faces, Using Text in Multimedia; Computers and Text, Font Editing and Design Tools, Hypermedia and Hypertext; Sound-Power of Sound, Multimedia Systems Sound, Digital Audio, Making MIDI Audio, Audio File Formats, MIDI vs Digital Audio, Sound in Multimedia Applications, Music CDs and Audio Production Guidelines. Still Images and Vector Graphic, Bitmaps, Vector Drawings, 3-D Drawing and Rendering, Colr and Image File Formats; Animation-Power, Principles, Techniques, File Formats, Devoping Animation; Video-Using Video, Working of Videos, Analog Standards, NTSC, PAL, SECAM and ATSC DTV; Digital Display Standards; Digital Video, Video recording etc; Shooting, Editing Video; Storyboarding, Platform, Lighting, Chroma Keys etc; and Optimizing Video File Storages.
3. **Multimedia Hardware and Software:** Multimedia Platforms, Connections- SCSI,IDE,USB and Firewire; Multimedia Storage Devices, MM Input-Output Devices, Multimedia Communication Devices; Multimedia Software Tools – Text, Editing and WP tools, OCR Software, Drawing and Painting Tools, 3-D Modeling and Animation Tools; Image and Sound Editing Tools; Animation, Video and Digital Movie Tools; and Multimedia Accessories.
4. **Multimedia Authoring and Integration:** Developing Multimedia Applications, Types of Authoring Systems: Object Based, Icon Based, Page based, Card-based, Stages of Authoring, Editing, Organizing, Interactivity, Performance Tuning, Cross Platform Features, Cross Platform Authoring Notes; Authoring Notes; Introduction to MX Flash MX / Director MX.
5. **Multimedia for WWW**: Internet, Internetworking, Bandwidth Issue, Internet Services, WWW and HTML, Web Pages, Dynamic Webpages and XML; Multimedia Web; Web Servers, Browsers and Search Engines; Web Page Makers and Site Builders, Plug-in and Delivery Vehicles; Beyond HTML, 3D Worlds; Multimedia on the Web- Workspace, Nibbling, HTML and multimedia; Text for Web, Images for the web – GIF, PNG Images, JPEG, Image Maps; Sound and Animation on the Web. Miscellaneous topics and supplements

**BOOKS**


1 Vaughon: Multimedia – Making it Work, Tata McGraw Hill
2 Parekh: Principles of Multimedia, Tata McGraw Hill
3 Li & Drew: Fundamentals of Multimedia, Pearson Education
4 Rao: Multimedia Communication Systems – Techniques, Standards and Networks

| DCA14 | Multimedia Applications Lab Part (Flash) | 0-0-2 | - |
|---|---|---|---|

**Implementation of at least ONE specific assignment concerning each of the following:**

1. Graphics ,animated objects, Flash assets and Interfaces
2. Navigation, Color dynamics and Graphic Objects
3. Basic animation techniques and typical Frame-by-Frame Animation
4. Symbols, Instances, and Library Assets: Tweening, Libraries, Symbols and Instances
5. Imported Artwork & Video: Motion Tweening, Masking & Bitmaps
6. Small Movie Clips and Buttons
7. Interaction by Scripts with ActionScript Language
8. Interaction with ActionScript
9. Movie and Imported Artwork & Video
10. Publishing and Uploading Flash files eg. http://webdesign.microteksol.com/Flash/

+

| DCA15 | Networking Technologies | 3-1-0 | 4 |
|---|---|---|---|

1. **Introduction to Networks Standards & Model:** Introduction to Computer Networks; Communication Media and Nodes; Workstations; Hosts and Servers; Packets, Frames, and Cells; Networking Capabilities; Peer-to-Peer Networking and Workgroups; Networking with Servers; Client-Server Networking; Local Area Network (LAN), Metropolitan Area Network (MAN), Wide Area Network (WAN), Enterprise Network; Networking Standards and their Types; ISO-OSI Model; TCP/IP Model.
2. **Topologies, Communication Media and Networks:** Network Topologies, Communication Media, Communication Media Costs and Considerations; Ethernet and the IEEE 802.3 Standards, Token Ring and the IEEE 802.5 Standards. WAN and Enterprise Network Communications; Fast Ethernet; FDDI; X.25, ISDN, Frame Relay; Multistation Access Units (MAU); Multiplexers, Repeaters, Bridges, Routers, Hubs, Gateways ; ATM Switches, VLANs.
3. **Network Planning and Fault-Tolerance Techniques:** Assessing Network Needs; Developing a Plan; Selecting the Right Network Media and Topology; Estimating Network Costs; Managing Network Performance through Centralized Planning; Planning Network Security; Component Failures; Fault Tolerance; Disk-Storage Fault Tolerance; Server Fault Tolerance; Developing a Disaster Recovery Plan.
4. **Remote Network Access and Network Monitoring:** A Brief History of Remote Access; Modems; Remote Access Protocols; Security; An Overview of Network Monitoring; Establishing Network Traffic Characteristics; SNMP; Network Monitoring Devices; An Overview of Enterprise Network Management Tools; Event Management; Domain Services Management; Server and Workstation Management; Firewalls.
5. **Troubleshooting Network Problems:** Developing a Problem Solving Strategy; Know Your Network; Know the Business Processes of Your Organization, Solving Problem Step-by-Step. Miscellaneous topics and supplements

**BOOKS**


1. Palmer: Hands-on Networking Essentials with Projects,Vikas Publishing House
2. Tanenbaum, Computer Networks, Pearson Education
3. William Stallings, Data and Computer Communications, PHI

| DCA16 | Lab-I (Programming in C) | 0-0-4 | 2 |
|---|---|---|---|

**Implementation of at least ONE specific assignment concerning each of the following:**

1. Simple programs based on operators and input/ output.
2. Programs based on conditional control statements.
3. Programs based on looping and transfer statements.
4. Programs based on functions.
5. Programs based on recursive functions.
6. Programs based on array.
7. Sorting and searching programs.
8. Programs based on structures.
9. File handling programs.
10. File handling with structure programs.

+

| DCA17 | Lab-II (Oracle) | 0-0-4 | 2 |
|---|---|---|---|

**Implementation of at least ONE specific assignment concerning each of the following:**

1. SQL statements to create, update, and delete databases and tables
2. SQL statements to insert, update, and delete records from tables
3. SQL statements to create, update, and delete views
4. Simple SQL queries to retrieve information from a database
5. Nested SQL queries to handle complex information retrieval requirements.
6. SQLqueries using aggregate functions like count, average, sum, etc.
7. PL/SQL blocks using basic data types and operators
8. PL/SQL blocks using branching and looping constructs
9. Database triggers using PL/SQL
10. Database functions/procedures using PL/SQL

**********

| DCA21 | Visual Programming | 3-1-0 | 4 |
|---|---|---|---|

1. **VB.NET Fundamentals:** Windows Applications; Programming Languages: Procedural, Event Driven, Object Oriented and Visual Paradigms; Object Model and MS-Visual Studio; Writing VB Projects; Programming VB vs other Projects; Visual Studio Environment; Typical Errors and Visual Studio Help Features.
2. **Controls, Declarations and Calculation:** Controls and their Importance; Multiple Controls: selection, Properties, Alignments etc; Designing GUI: Events, KAK, Defaults, Tab orders and Yool-tips; Coding for Controls and Programming; Data, Variables and Constants; Calculations; Formatting; Handling Exceptions; and Ménage Boxes. Conditions; If and nested 1F statements; Radio Buttons and Text Boxes; Enhancing Message Boxes; Input Vilifications and VB-Based Provisions; Calling Event Procedures; and Debugging VB Projects; Minus and Common Dialog Boxes; Creating Context Minus; and Writing General Sub-procedures and Functions.
3. **Object Oriented Programming:** OOP Concepts: Objects, 0objet 0rientation Attributes and Terminology; Reusable Objects; Multi-tier Applications; Classes; Creating New Objects; Constructors and Destructors; Inheritance; Object Browser; List Boxes and Combo Boxes; Do/Loops; for/next loops; Selection of Entries and Printing; Programming Example; Arrays; Case Structure, Sharing Event Procedures, 1-Dim Arrays, for/Next Statements; Structures; Accumulators; Table Looking; List Boxes with Arrays; Multi-Dimensional Arrays; and a Programming example.
4. **Data Storage and Retrievals:** DB Files; File vs Arrays, DB terminology and XML data; Using ADO.NET and VB; Creating DB Applications; Using Data-Bound Labels; Populating Combo Boxes with Data; Making DB objects Portable; Updating Data Sets; and Programming example. Date File and Project Files; Data File Terminology, Using Streams and File Handling, Using the File Common Dialog Boxes; Saving Contents of List Boxes; Serialization – Making a Class Serializable, Adding Formatter Object File Stream, Saving and Stream and Recreating Objects; and a programming example.
5. **Web Form, Graphics and Animation:** VB and the Web Programming; Client Server Web Application; Creating Web Forms – WF in Visual Studio IDE; HTML Code, Browser view, Toolbars Controls, Event handling, Debugging etc; Laying out Web Forms; Validator Controls, Managing Web Projects and Programming Web Forms. Graphics in Windows and on the web; The graphics Environment Drawing Objects, Paint Procedure; Pen and Brush objects, Coordinate system, and Graphics Methods; Random Numbers; Animation – Display, Control and Movements; The Timer Control; Scrool Bar Controls: Properties, Events and Programming; and a Programming Example. Miscellaneous topics and supplements

**BOOKS**


1. **Bradley et al.: Programming with VB.NET, McGraw Hill**
2. Deitel & Deitel: VB.NET – How to Program?, GadPilan
3. Date: Programming and Problem Solving with VB.NET, Narosa Publishing House
4. Robin: Object Oriented Programming with VB.NET, PHI


| DCA22 | Operating Systems and Shell Programming | 3-1-0 | 4 |
|---|---|---|---|

1. **Introduction:** Operating System Definition; Operating System Structure; Operating System Operations; Process Management; Memory Management; Storage Management; Protection and Security; Distributed Systems; Special-Purpose Systems; Computing Environments; Open Source Operating Systems. Operating System Services; User Operating System Interface; System Calls; Types of System Calls; System Programs; Operating System Design and Implementation; Operating-System Structure; Virtual Machines, Operating System Generation; System Boot.
2. **Process Management:** Introduction; Process Scheduling; Operations on Processes; Process Scheduling Criteria, Scheduling Algorithms, Operating System Examples; Process Synchronization

– Introduction; Critical-Section Problem; Peterson's Solution; Synchronization Hardware; Semaphores; Classic Problems of Synchronization; Monitors; Synchronization Examples; Atomic Transactions. Deadlocks – System Model; Deadlock Characterization; Methods for Handling Deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection and Recovery.

3. **Memory Management:** Basic Concepts; Swapping; Contiguous Memory Allocation; Paging; Structure of Page Table; Segmentation; Virtual Memory; Demand Paging; Copy-on-Write; Page Replacement; Page Replacement Algorithms; Allocation of Frames; Thrashing; Memory-Mapped Files; Allocating Kernel Memory.
4. **Basic Concepts of UNIX:** The Unix Environment, Unix Structure, Commands, The vi Editor, Modes, Commands; File Systems – File Names, File Types, Regular Files, Directories, File System Implementation, Operations on Directories and Files.
5. **Security and File Permission:** Users and Groups; Security Levels; Changing Permissions; User Masks; Changing Ownership and Group, Introduction to Shells; UNIX Session; Standard Streams; Redirection; Pipes; Command Execution; Command-Line Editing; Quotes; Command Substitution; Job Control; Aliases; Variables; Predefined Variables; Options; Shell/ Environment Customization. Miscellaneous topics and supplements

**BOOKS**


1. Silberschatz, Galvin and Gagne: Operating System Concept, John Willey
2. Behrouz A. Forouzan: Unix and Shell Programming, Thomson, Indian Reprint
3. Stallings, W.: Operating Systems – Internals and Design Principles, Pearson Education
4. Sumitabha Das: UNIX – Concepts and Applications.


+

| DCA23 | Website Design and Management | 3-1-2 | 4 |
|---|---|---|---|

1. **Introduction to Web Sites:** Overview; Technical Infrastructure; Information Sharing; Types of Web Sites; Website Architecture – 2-Tiered and n-Tiered Architecture, Website Design Issues – Administrative, Creative, Technical, and Marketing; Introduction to Web Site Editors. Web Site Design Process – Analysis, Design, Development, Implementation, Evaluation and Maintenance, and Promotion; Factors Influencing Web Site Design; Elements of Web Site Design, Web Page Design and Layout.
2. **Web Page Design**: Introduction; Components of a Web Page – Typography (Fonts and Styles), Color (Color Models, Color Profiles, etc.), Graphics (Raster and Vector Graphics), Audio, Video, Plug-ins; Slicing Images; making Images Load Faster; Reducing the Colors in an Image; Reducing the Size of Images – Resizing, Cropping, and Creating Thumbnails; Developing Web Site Using Dreamweaver, Testing a Web Site.
3. **Hyper Text Mark-up Language:** HTML Document Structure – HTML, HEAD, and BODY; HTML Tags and their Functions; Creating Web Pages Using HTML. Dynamic HTML (DHTML): Introduction, DHTML Features – Dynamic Content, Dynamic Style, Dynamic Positioning, Data Binding; Components of DHTML – Cascading Style Sheets (CSS); Animations; Types of Animations – GIF Animations, Swf Animations, DHTML Animations, Web Video Animations, 3-Dimentional and Virtual Reality Animations; Object Referencing, Dynamic Styles, Dynamic Positionin**g**, Event Model – Event onclick, Event onload; Filters and Transition – Flip Filters (flipv and fliph) ,Transparency with Chroma Filter, Creating Image Masks, Miscellaneous Image Filters (invert, gray and X-ray).
4. **Java Script:** Introduction to Scripting; Obtaining User Input with Prompt Dialogs, Memory Concepts, Arithmetic; Control Statements& Loop structures, Break and Continue Statements, Labeled Break and Continue Statements. Functions – Programmer Defined Functions, Function Definition, Scope Rules, Java Script Global Functions, Recursion. Arrays –Declaring, Allocating

and using Arrays; Sorting Arrays; Searching Arrays; Multidimensional Arrays Introduction to Objects; Types of Objects – Math, String, Date, Boolean, and Window Objects. ASP.NET – Introduction, .NET Overview, Setup, JScript.NET, Web Forms, Session Tracking.

5. **Web Servers**: Introduction to Internet Information Services (IIS), HTTP Request Types, System Architecture, Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers; Apache Web Server; Requesting Documents (XHTML, ASP.NET, Perl, PHP, Python). Miscellaneous topics and supplements

**BOOKS**


1. Basics of Website Design, PHI
2. Deitel – Internet & World Wide Web: How to Program, PHI
3. Steven Holzner - HTML Black Book, DreamTech Press
4. Michael K. Glass, et. al.- Beginning PHP, Apache, MySQL Web Development, Wrox


+

| DCA24 | Object Oriented Programming using C++ | 3-1-0 | 4 |
|---|---|---|---|

1. **Introduction**: Programming Paradigms: Unstructured Programming, Structured Programming, Object Oriented Programming; Abstract Data Type (ADT); Class; Object; Message; Encapsulation; Polymorphism; Inheritance; Pros and Cons of Object Oriented Methodology; cin and cout Objects.
2. **Classes and Objects:** Classes; Friend Functions: Introduction, Benefits and Restrictions, Friends Classes; Inline Functions; Constructor: Introduction, Parameterized Constructor; Destructor and its usages; Static Data Member and Static Member Functions; Creating Object; Passing and Returning Object(s) to/from a Function; Object Assignment; Nested and Local Classes; Arrays of Objects; Pointer to Objects; this Pointer, Pointer to Derived Type; References; Reference vs Pointer; Reference Parameters; Dynamic Memory Allocation: new and delete Operators.
3. **Function and Operator overloading**: Function overloading: Introduction, Rules, Overloading Constructors, Copy Constructors; Default Function Arguments vs. Function Overloading. Operator Overloading: Introduction, Operators that cannot be Overloaded, Overloading Operators using Member Function and Friends Functions, Overloading different operators including prefix and postfix form of ++ and -- operators, Shorthand Operators, new, delete, [], (), -> and comma Operators.
4. **Inheritance & Virtual function**: Inheritance: Introduction, Types of Inheritances, Base-Class Access Control, Protected Members, Protected Base-class Inheritance, Multiple Inheritance, Problem in Multiple Inheritance, Solution to Multiple Inheritance Problem, Passing Parameters to Base Class Constructors; Virtual functions: Introduction, Calling a Virtual Function using Base Class Reference, Pure Virtual Function, Abstract Class.
5. **Generic Function, Exception and File Handling:** Generic Functions: Benefits, Functions with Two Generic Types, Explicitly Overloading a Generic Function, Overloading a Function Template, Restriction, Generic Sort, Generic Class. Exception Handling: Introduction, Using try and catch Blocks, Creating Exception Class, throwing Object. C++ Streams; C++ File Handling: Opening and Closing a File, Reading and Writing a Text File, Random Access, Reading and Writing Object to a File

**BOOKS**


1. Herbert Schildt: Complete Reference C++, 3rd Ed., Tata Mc Graw Hill
2. BJarne Stroustrup: The C++ Programming Language, Pearson Education
3. H.M. Deitel & P.J. Deitel: C++ How to Program, 4th Ed., Pearson Education.

| DCA25 | Lab-III (VB .NET) | 0-0-4 | 2 |
|---|---|---|---|

**Implementation of at least ONE specific assignment concerning each of the following:**

1. Common Controls, Form Design, setting Properties and Perform Arithmetical and Logical operations.
2. Decision making, arrays and loop statements eg. Prime/Leap year/ etc. without readymade function.
3. Advanced controls i.e. List box, Combo box, MDI etc.
4. Menus and sub menus Context menu for various operations i.e. Arithmetic operations.
5. Sorting techniques and MATRIX operations.
6. Console operations and operations with Debugging Window
7. Classes, Objects, Inheritance etc.
8. File operations eg Writing, Reading, Copying etc.
9. Graphics for Drawing Line, Rectangle, Circle, FillEllipse, Flowers, RGB etc.
10. Database Connectivity with some Databases i.e. MS ACCESS, SQL Server, Operations.

+

| DCA26 | Lab-IV (Programming in C++) | 0-0-4 | 2 |
|---|---|---|---|

**Implementation of at least ONE specific assignment concerning each of the following:**

1. Programs based on data types, input and output.
2. Programs based on class.
3. Programs based on function overloading.
4. Programs based on friends functions.
5. Programs based on operator overloading.
6. Programs based on default arguments.
7. Programs based on Inheritance.
8. Programs based on virtual functions and function overriding.
9. Programs based on template functions and classes.
10. Programs based on Exceptions and file handling with file stream.

# Annexure – II

# Ph.D. Programme Course Work Syllabus

| Sr No | Course Code | Course Title | Marks (I.A. + Sem) |
|---|---|---|---|
| 1 | PhDCW01 | Research Methodologies in Computer Science and Information Technology (RMCSIT) | 100 (25 + 75) |
| 2 | PhDCW02 | Research Challenges in Computer Science and Information Technology (RCCSIT) | 100 (25 + 75) |
| 3 | PhDCW03 | Annotated Bibliography and Literature Review of Research Papers (ABLRRP) | 50 (15 + 35) |

## Course I

### PhDCW01: Research Methodologies in Computer Science and Information Technology (RMCSIT)

UNIT I: **Fundamentals of Research**

Meaning and Idea of Research, Motivation of Research, Research and Scientific Methods, Developing Attitude and Aptitude for Research, Issues and Intricacies Involved, Research Characteristics and Scientific Methods, Role of Literature Survey, Ways to Perform Literature Survey, Methods to find open problem and Research Problems, Research Process and Stages of Research, Defining the Research Problem, Formulation of Research Questions, Criteria of Good Research, High Quality Research, Sources of Research Problems.

UNIT II: **Peripheral Issues of Research**

Role of Data, Facts, Information and Knowledge, Events, Information and Data Interpretation, Data Analysis and Different Research Design, Ethics in Research, Role of Critical Thinking and observational skills, Creative Thinking in Research Design and Development, Purpose of Quantification and Validation, Rationality of Research Outcome, Learning Outcomes, Expectation from Research Scholars in Indian and Global Scenario, Problems Encountered by Researchers in India, Nature and Types of Research in Computer Science and Information Technology.

UNIT III: **Uncertainties, Methodologies, Tools & Techniques for Research**

Introduction to Uncertainties, Types of Uncertainties, Randomness, Measurements and Quantification of Uncertainties, Complexities, ambiguity, cloudiness, etc, Probabilities, Random variables, Descriptive Statistics, Correlation and Regression Analysis, Sampling Distribution and Probability Distribution, Hypothesis Testing, Interpolation and Extrapolation, Multivariate Analysis, Fuzzy Set and System, Fuzzy Logic, Fuzzy Intervals, Fuzzy Categories, Fuzzy Relation Equations, Rough Set, Fuzzy and its Application in Bioinformatics, Entropy and Its Applications, Multiscale Entropy Algorithm and Its Variants, Applications of Maximum Fuzzy Entropy Principle, Method of Constructing Fuzzy Entropies, Fuzzy Entropy and Fuzzy Subsethood Measure.

UNIT IV: **Academic Ethics and Technical Writing**

Academic Writings and Paper Clinic, Significance of Literature Review, Writing Scientific Report, Structure and Components of Research Report, Revision, Writing Project Proposal, Items of Research Proposals, Writing Research Paper, Citation Counting and Impact Factor, Science Citation Index(SCI), Science Citation Index Expanded(SCIE), H-Index, Academic Ethics and Plagiarism, Exposures on Plagiarized software, Intellectual Property Rights and Patent Law, Digital Ethnography.

UNIT V: **Study and Practice of Advance Tools in Computer Science**

Exposure and Experimentation with Case Studies and Minor Projects with their Implementation using Tools and Techniques covered in Unit III on platforms / languages such as MATLAB, Perl, Latex, R-Language, NS-2, Multimedia Tool (MAYA) /Macromedia Flash Professional 8, SPSS, E-Views, Simulations, and any other advance tools of future to be adopted as and when desired.

References:


1. Leedy P. D. and Ormrod J. E., Practical Research: Planning and Design, 7th Edition. Prentice Hall, 2001.
2. Chicago University Press, "The Chicago Manual of Style: The essential Guides for Writers, Editors and Publishers", 16th Edition, Edited by University of Chicago Staff, London, 2010.
3. Larry B. Christensen et all "Research Methods, Design, and Analysis", 12th Edition, Pearson Publications, 2014.
4. Kothari, C R, "Research Methodology: Methods and Techniques", New Age International Publications, New Delhi, 2004.
5. John Ellison Kahn, "How to Write and Speak Better", Reader's Digest Association Ltd., USA, New York, 2006.
6. Sarah Pink et all, "Digital Ethnography: Principles and Practice", SAGE Publications Ltd., London, 2016.

Web reference: https://owl.english.purdue.edu/sitemap/


**********

## Course II

### PhDCW02: Research Challenges in Computer Science and Information Technology (RCCSIT)

UNIT I: **Algorithm, Architectures, Programming Paradigms and Legacy System**

Nature and Types of Research in Computer Science and Information Technology, Review of Various Strategies used for Analysis and Designing of Algorithms, Efficiency of Algorithms, Advance Algorithms, P versus NP problem, Better algorithm for Matrix Multiplication and Fourier Transformation, Building Quantum Computers using Shor's Algorithm, Advance Architecture, Nano Computers and Modelling, various approaches of Software Development and its Optimization with Qualitative Improvements of the Process, Process Optimization and effectiveness, verification and validations, SQC and SQA, Various Quality Standards, Issues and Challenges of Legacy System Developments and its parameters.

UNIT II: **Computer Network, Security, Social Media and Digital Democracy**

Advance Computer Architectures, Growing Complexity of Computer Network and Security Challenges, Distributed Database, Security and Cryptography, Quantifications and assessments of Abstract Concepts, such as emotions, Capturing of Emotions and Feelings and its Interpretations in Social media and in Digital Democracy, Multimedia graphics audio and video analysis, comprehending movement, synthesizing realistic multimedia, Cyber Laws and its Challenges.

UNIT III: **A.I. and its Applications - I (Business Intelligence, DM&DW, Business Expert Systems, Neural Network (NN), Natural language Processing (NLP), Image Processing and Pattern Matching (IP&PM), Speech and Voice Recognition (SVR), Human Computer Interaction (HCI))**

Overview of Data Mining and data warehousing, its Applications, including in medical and marketing researches, Development of Business Expert System and issues involved, State-of the-Art of Research in Pattern Matching and Image Processing, Recent Advancements in NLP, Speech and Voice Recognition, Human computer Interaction, Development of Answering Machine, Designing of Wearable Device, Use of AI techniques as Translational Research, in Management and Human Development, Emerging Technologies in Emerging Economies, GRAIN in BRICS.

UNIT IV: **A.I. and its Applications – II (Application of Neural Network, Machine learning, and other A I techniques in Bioinformatics, Computational and System Biology, Genetic Engineering, Biomedical Engineering, Drug Design, Computer Aided Design, Device Management, Robotics Surgery)**

Fundamentals of neural network, use of NN, other AI techniques in Bioinformatics and in Computational Biology, System Biology including Drug Design and Biomedical Engineering, Real life Case Studies and Implementation on Computer Aided Design, Innovations in Designing of Devices for Medical Treatments, Robotics and its Application in Operation theatre, Automobile Manufacturing Industries and in war fares. Applications of Robotics in Control and Monitoring of Illegal Activities against Human Being, such as Drug trafficking, Terrorism network, Flow of Black Money from India to Abroad, Synchronized Learning from various International Data Banks.

UNIT V: **Internet of Things (IOT) and Cloud Computing**

Introduction of IOT, Embedded Systems, Networking Over Embedded Devices, Cloud Computing, Different Architectures of Cloud Computing Issues and Challenges, Web Technologies and Web based developments, Building Smart Devices: internetworking of physical devices, vehicles, buildings embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data, M2M connectivity, cyber-physical systems, smart grids, smart homes, intelligent transportation and smart cities.

**References:**

**Unit I:**


1. Thomas H. Cormen, et al, "Introduction to Algorithms", Third Edition, Prentice Hall India Learning Private Limited, 2010.
2. Donald E. Knuth, "The Art of Computer Programming Volumes 1-4A", Addison Wesley, 2011.
3. Neapolitan and Naimipour, “Foundations of Algorithms using C++ Pseudo Codes”, JBL Publications, 2016.
4. Vishal Sahni, "Quantum Computing", McGraw Hill Education, 2007.
5. Roger S. Pressman, "Software Engineering: A Practitioner's Approach", Seventh Edition, McGraw Hill Education, 2014.
6. Saikut Dutt, "Software Engineering", Pearson Education India, 2015.]


**Unit II:**


1. Andrew S. Tanenbaum, et al, "Computer Networks", Fifth Edition, Pearson Education India, 2013.
2. Jeffery S. Beasley, "A Practical Guide to Advanced Networking", Third Edition, Pearson IT Certification, 2012.]
3. Stalling, "Cryptography and Network Security: Principles and Practice", Sixth Edition, Pearson Education India, 2013.
4. Bruce Schneier, "Applied Cryptography", Wiley, 1996.
5. Peter Mika, "Social Networks and the Semantic Web (Semantic Web and Beyond)", 2007 Edition, Springer, 2007.
6. Steinmetz, "Multimedia: Computing Communications & Applications", First Edition, Pearson Education India, 2002.


**Unit III:**


1. Stuart Russel and Peter Norvig: Artificial Intelligence–A Modern Approach, 3rd Ed., 2012, Pearson Education, ISBN: 0-13-790395-2
2. Elaine Rich, Kevin Knight, Nair: Artificial Intelligence, 2009, Tata McGraw Hill, 3rd Ed, ISBN-10: 0-07-008770-9
3. B. YegnaNarayana: Artificial Neural Network, EEE, PHI, 2001, ISBN 81-203-1253-8
4. Tom M. Mitchell: Machine Learning, McGraw-Hill, International Edition 1997, ISBN 0-07-115467-1
5. Ivan Bratko: PROLOG Programming, 3rd Ed., 2001, Pearson Education, ISBN: 81-7808-257-8
6. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publisher
7. Simon Haykin, Neural Networks And Learning Machines (3rd. Edition), 2012, PEd.
8. Kalyanmoy Deb (1999), research papers on Multi-objective genetic algorithms

9. Han, "Data Mining: Concepts and Techniques", Third Edition, Elsevier, 2007.
10. Christopher Bishop, "Pattern Recognition and Machine Learning (Information Science and Statistics), Springer, 2010.]
11. Jurafsky, et al, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Second Edition, Pearson, 2013.

**Unit IV:**


1. Tom M. Mitchell, "Machine Learning", First Edition, McGraw Hill Education, 2013.
2. John Hearty, "Advanced Machine Learning Python", Packt Publishing Limited, 2016.]
3. Saeed B. Nikku, "Introduction to Robotics: Analysis, Control, Applications", Second Edition, Wiley, 2011.
4. Neural Networks and Learning Machines, Third Edition, Simon Haykin McMaster University, Canada
5. Machine Learning, Tom Michael Mitchell ,McGraw-Hill Education
6. Artificial Intelligence A Modern Approach-third Edition by Stuart Russel, PHI
7. Computational Systems Biology , by Andres Kriete (Editor), Roland Eils (Editor)
8. Computational Molecular Biology, Book by Pavel A. Pevzner


**Unit V:**


1. David Boswarthick, et al, "The Internet of Things: Key Applications and Protocols", Wiley, 2015.
2. Erl, "Cloud Computing: Concepts, Technology & Architecture", First Edition, Pearson Education India, 2014.


**Note**: Without Loss of Quality, Text and References may be updated further as and when desired at the time of Executions / Implementations for both the Courses.

**********

# *Question Paper Format*

## *MCA & PGDCA Semesters, Examinations*

**Code:** paper-code Roll No……..

**Programme Name (SEM) EXAMINATIONS - 2016**

Paper Title

**Time: 2 Hours** **Max Marks: 75**

- *Write your Roll No. on the top immediately on receipt of the question paper.*
- *Attempt ALL questions by selecting any TWO parts. All questions carry equal marks.*

1. **(a)**

   **(b)**

   **(c)**

2. **(a)**

   **(b)**

   **(c)**

3. **(a)**

   **(b)**

   **(c)**

4. **(a)**

   **(b)**

   **(c)**

5. **(a)**

   **(b)**

   **(c)**

❖ ❖ ❖

Note: The questions 1 to 5 will be sets from unit 1-5 respectively.

# Question Paper Format

# Ph.D.(Pre Ph.D. Course Work) Examinations-2016

**Code:** paper-code Roll No……..

**Programme Name (SEM) EXAMINATIONS - 2016**

Paper Title

**Time: 2 Hours** **Max Marks: 75**

- *Write your Roll No. on the top immediately on receipt of the question paper.*
- *Attempt ALL questions by selecting any TWO parts. All questions carry equal marks.*

1. **(a)**

   **(b)**

   **(c)**

2. **(a)**

   **(b)**

   **(c)**

3. **(a)**

   **(b)**

   **(c)**

4. **(a)**

   **(b)**

   **(c)**

5. **(a)**

   **(b)**

   **(c)**

Note: The questions 1 to 5 will be sets from unit 1-5 respectively.

(**Dr. Syed Afzal Murtaza Rizvi)**

***Professor and Head***

# JAMIA MILLIA ISLAMIA

(A Central University by an Act of Parliament)

## DEPARTMENT OF COMPUTER SCIENCE

Maulana Mohammad Ali Jauhar Marg, New Delhi-110025
Tel.: 26985177, 26980014, 26981717, EPABX : 3450, 3452


Date: October 23,2019

It is hereby informed that by introducing courses focus on employability/ entrepreneurship/ skill development in the curriculum of Master of Computer Applications(MCA ) and Post Graduate Diploma in Computer Applications( PGDCA), Department of Computer Science w.e.f. 2016 onwards students' placement has been increased.

| Name of the Course | Course Code | Name of the Programme |
|---|---|---|
| MATLAB Computation | CBCSS118 | Master of Computer Applications |
| Multimedia with Flash | CBCSS228 | Master of Computer Applications |
| Mobile Applications Development | CBCSS338 | Master of Computer Applications |
| Programming in Python | CBCSS448 | Master of Computer Applications |
| Big Data Analytics with HADOOP | CBCSS558 | Master of Computer Applications |
| DTP with MS Office | CBCSS108 | Post Graduate Diploma in Compute Applications |
| Website Design and Maintenance | CBCSS208 | Post Graduate Diploma in Compute Applications |
| Digital Image Processing and GPU Programming | CSCC51/CBCS51 | Master of Computer Applications |

Professor & Head
Department of Computer Science
Faculty of Natural Sciences
Jamia Millia Islamia

# JAMIA MILLIA ISLAMIA
(A Central University by an Act of Parliament)

## DEPARTMENT OF COMPUTER SCIENCE
Maulana Mohammad Ali Jauhar Marg, New Delhi-110025
Tel.: 26985177, 26980014, 26981717, EPABX : 3450, 3452


| | | |
|---|---|---|
| Data Mining and Data Warehousing | CSCC52/CSAE52 | Master of Computer Applications |
| Machine Learning and Soft Computing | CSCC53 | Master of Computer Applications |
| Pattern Matching using Python Programming | CSCC54 | Master of Computer Applications |
| BIG DATA Analytics and Cloud Computing | CSCC44 | Master of Computer Applications |
| Leadership, Interpersonal and Group Dynamics | CSCC45 | Master of Computer Applications |
| MATLAB Computation | CBCSS118 | Master of Computer Applications |
| Mobile Applications Development | CBCSS338 | Master of Computer Applications |
| Information Security | CSC552 | Master of Computer Applications |

**Prof. S.M.K.Quadri**
Head
Department of Computer Science

Head of Dept.
Department of Computer Science
Faculty of Natural Sciences
Jamia Millia Islamia
New Delhi-110025