

॥ सा विद्या या विमुक्तये ॥



# स्वामी रामानंद तीर्थ मराठवाडा विद्यापीठ, नांदेड

“ज्ञानतीर्थ” परिसर, विष्णुपुरी, नांदेड - ४३१६०६ (महाराष्ट्र)

**SWAMI RAMANAND TEERTH MARATHWADA UNIVERSITY NANDED**

“Dnyanteerth”, Vishnupuri, Nanded - 431606 Maharashtra State (INDIA)

Established on 17th September 1994 – Recognized by the UGC U/s 2(f) and 12(B), NAAC Re-accredited with 'A' Grade



## ACADEMIC (1-BOARD OF STUDIES) SECTION

Phone: (02462) 229542

Website: www.srtmun.ac.in

E-mail: bos.srtmun@gmail.com

Fax : (02462) 229574

संलग्नित महाविद्यालयांतील विज्ञान व तंत्रज्ञान विद्याशाखेतील पदव्युत्तर स्तरावरील प्रथम वर्षाचे CBCS Pattern नुसारचे अभ्यासक्रम शैक्षणिक वर्ष २०१९-२० पासून लागू करण्याबाबत.

### प रि प त्र क

या परिपत्रकान्वये सर्व संबंधितांना कळविण्यात येते की, दिनांक ०८ जून २०१९ रोजी संपन्न झालेल्या ४४व्या मा. विद्या परिषद बैठकीतील ऐनवेळचा विषय क्र.११/४४-२०१९ च्या ठरावानुसार प्रस्तुत विद्यापीठाच्या संलग्नित महाविद्यालयांतील विज्ञान व तंत्रज्ञान विद्याशाखेतील पदव्युत्तर स्तरावरील प्रथम वर्षाचे खालील विषयांचे **C.B.C.S. (Choice Based Credit System) Pattern** नुसारचे अभ्यासक्रम शैक्षणिक वर्ष २०१९-२० पासून लागू करण्यात येत आहेत.

1. Bioinformatics
2. Biotechnology
3. Biochemistry
4. Botany
5. Chemistry
6. Computer Management
7. Computer Science
8. Dairy Science
9. Environmental Science
10. Herbal Medicine
11. Information Technology
12. M.C.A.
13. Microbiology
14. Physics
15. Software Engineering
16. System Administration & Networking
17. Zoology

सदरील परिपत्रक व अभ्यासक्रम प्रस्तुत विद्यापीठाच्या [www.srtmun.ac.in](http://www.srtmun.ac.in) या संकेतस्थळावर उपलब्ध आहेत. तरी सदरील बाब ही सर्व संबंधितांच्या निदर्शनास आणून द्यावी.

‘ज्ञानतीर्थ’ परिसर,

विष्णुपुरी, नांदेड - ४३१ ६०६.

जा.क्र.: शैक्षणिक-१/परिपत्रक/पदव्युत्तर-सीबीसीएस  
अभ्यासक्रम/२०१९-२०/४६४

दिनांक : ११.०७.२०१९.

प्रत माहिती व पुढील कार्यवाहीस्तव :

- १) मा. कुलसचिव यांचे कार्यालय, प्रस्तुत विद्यापीठ.
- २) मा. संचालक, परीक्षा व मूल्यमापन मंडळ यांचे कार्यालय, प्रस्तुत विद्यापीठ.
- ३) प्राचार्य, सर्व संबंधित संलग्नित महाविद्यालये, प्रस्तुत विद्यापीठ.
- ४) साहाय्यक कुलसचिव, पदव्युत्तर विभाग, प्रस्तुत विद्यापीठ.
- ५) उपकुलसचिव, पात्रता विभाग, प्रस्तुत विद्यापीठ.
- ६) सिस्टम एक्सपर्ट, शैक्षणिक विभाग, प्रस्तुत विद्यापीठ.



स्वाक्षरित/—

**उपकुलसचिव**

शैक्षणिक (१-अभ्यासमंडळ) विभाग

**Swami Ramanand Teerth Marathwada  
University, Nanded**  
(NAAC Re-accredited with 'A' Grade)



**Syllabus of**  
**M.Sc. (Software Engineering) (2 years)**  
(Revised CBCS pattern)

**Introduced from Academic Year 2019-2020**

# M.Sc. Software Engineering

**M.Sc. Software Engineering** (2years) program / degree is a specialized program in software engineering and software development processes issues. It builds the student on higher studies and research awareness in overall designing and development of software so as to become competent in the current race and development of new computational sciences. The duration of the study is of four semesters, which is normally completed in two years.

## **CBCS pattern**

**The M.Sc. Software Engineering** program as per CBCS (Choice based credit system) pattern, in which choices are given to the students under open electives and subject electives. The students can choose open electives from the wide range of options to them.

## **Eligibility and Fees**

The eligibility of a candidate to take admission to **M.Sc. Software Engineering** program is as per the eligibility criteria fixed by the University. More details on admission procedure and fee structure can be seen from the prospectus of the college / institution as well as on website of the University.

## **Credit Pattern**

Every course has corresponding grades marked in the syllabus structure. There are 25 credits per semester. A total of 100 credits are essential to complete this program successfully. The Grading pattern to evaluate the performance of a student is as per the University rules.

Every semester has a combination of Theory (core or elective) courses and Lab courses. Each theory course has 04 credits which are split as 02 external credits and 02 internal credits. The university shall conduct the end semester examination for 02 external credits. For theory internal credit, student has to appear for 02 class test (15 marks) and 01 assignment (20 marks). Every lab course has 02 credits which are split as 01 external credit and 01 internal credit. For lab internal credit, the student has to submit Laboratory Book (05 marks) and remaining 20 marks are for the Lab activities carried out by the student throughout the semester. For lab external credit, 20 marks are reserved for the examinational experiment and 05 marks are for the oral / viva examinations. There is a special skill based activity of 01 internal credits per semester which shall inculcate awareness regarding the domain of computers, IT, and ICT.

The open elective has 04 credits which are purely internal. If students are opting for MOOCs as open elective, then, there must be a Faculty designed as MOOCs course coordinator who shall supervise learning through MOOCs. This is intentionally needed as the MOOCs course coordinator shall verify the MOOC details including its duration, starting date, ending date, syllabus contents, mode of conduction, infrastructure feasibility, and financial feasibility during start of each semester. This is precautionary as the offering of the MOOCs through online platforms are time specific and there must be proper synchronization of semester duration with the MOOCs duration. Students must opt for either institutional / college level open elective or a course from University recognized MOOCs platforms as open electives.

The number of hours needed for completion of theory and practical courses as well as the passing rules, grading patterns, question paper pattern, number of students in practical batches, etc shall be as per the recommendations, norms, guidelines and policies of the UGC, State Government and the SRTM University currently operational. The course structure is supplemented with split up in units and minimum numbers of hours needed for completion of the course, wherever possible.

Under the CBCS pattern, students would graduate **M.Sc. Software Engineering** with a minimum number of required credits which includes compulsory credits from core courses, open electives and program specific elective course. All students have to undergo lab / practical activities leading to specific credits and project development activity as a part of professional UG program.

1. **M.Sc. Software Engineering** Degree / program would be of 100 Credits. Total credits per semester= 25
2. Each semester shall consist of three core courses, one elective course, one open elective course and two practical courses. Four theory courses ( core+elective) = 16 Credits. Two practical / Lab courses= 4 Credits in total (02 credits each) , One Open elective= 4 credit, One skill enhancement activity of 01 credits.
3. One Credit = 25 marks , Two Credits = 50 Marks, Four Credits = 100 Marks

### PEO, PO and CO Mappings

1. **Program Name** : MSc.( SE)
2. **Program Educational Objectives:** After completion of this program, the graduates / students would

PEO I :Technical Expertise	Implement fundamental domain knowledge of core courses for developing effective computing solutions by incorporating creativity and logical reasoning.
PEO II : Successful Career	Deliver professional services with updated technologies in software engineer based career.
PEO III :Hands on Technology and Professional experience	Develop leadership skills and incorporate ethics, team work with effective communication & time management in the profession.
PEO IV :Interdisciplinary and Life Long Learning	Undergo higher studies, certifications and research programs as per market needs.

3. **Program Outcome(s):** Students / graduates will be able to

- PO1:** Apply knowledge of mathematics, science and algorithm in solving Computer problems.
- PO2:** Generate solutions by understanding underlying software engineering environment
- PO3:** Design component, or processes to meet the needs within realistic constraints.
- PO4:** Identify, formulate, and solve problems using computational temperaments.
- PO5:** Comprehend professional and ethical responsibility in computing profession.
- PO6:** Express effective communication skills.
- PO7:** Recognize the need for interdisciplinary, and an ability to engage in life-long learning.
- PO8:** Actual hands on technology to understand it's working.
- PO9:** Knowledge of contemporary issues and emerging developments in computing profession.
- PO10:** Utilize the techniques, skills and modern tools, for actual development process
- PO11:** Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings in actual development work
- PO12:** Research insights and conduct research in computing environment.

4. **Course Outcome(s):** Every individual course under this program has course objectives and course outcomes (CO). The course objectives rationally match with program educational objectives. The mapping of PEO, PO and CO is as illustrated below

## 5. Mapping of PEO& PO and CO

Program Educational Objectives	Thrust Area	Program Outcome	Course Outcome
PEO I	Technical Expertise	PO1,PO2,PO3,PO6	All core courses
PEO II	Successful Career	PO4,PO5,PO11,	All discipline specific electives courses
PEO III	Hands on Technology and Professional experience	PO8,PO10	All Lab courses
PEO IV	Interdisciplinary and Life Long Learning	PO7,PO9,PO12	All open electives and discipline specific electives

The detailed syllabus is as below,

Sr. No	Course category	Course Code	Course Title	Internal credits	External credits	Total credits
<b>First Semester</b>						
1.	<b>Core Subjects</b>	SE-101	Programming with C++	2	2	4
2		SE-102	Software Engineering	2	2	4
3		SE-103	Operating System Concepts	2	2	4
<b>Choose any one from below elective subjects</b>						
4	<b>Elective Subject</b>	SE-104 A	Computer System Security	2	2	4
		SE-104 B	Principles of Programming Language concept			
<b>Practical /Lab</b>						
5	<b>Lab / Practical</b>	SE-105	Lab1 : Programming in C++	1	1	2
		SE-106	Lab-2 : Based on Elective subjects	1	1	2
6	<b>Open Elective</b>	SE-107A	University recognized MOOC (NPTEL / SWAYAM / others) OR Intra / Inter Departmental OR Intra / Inter School OR	4	0	4
		SE-107 B	Research Methodology			
7	<b>Skill based Activity</b>	SE-108	SK-01	1	0	1
	Total credits					<b>25</b>

<b>Sr. No</b>	<b>Course category</b>	<b>Course Code</b>	<b>Course Title</b>	<b>Internal credits</b>	<b>External credits</b>	<b>Total credits</b>
<b>Second Semester</b>						
1.	<b>Core Subjects</b>	SE-201	Linux Operating System concepts and Administration	2	2	4
2		SE-202	Advance JAVA Programming	2	2	4
3		SE-203	Database Administration	2	2	4
<b>Choose any one from below elective subjects</b>						
4	<b>Elective Subject</b>	SE-204 A	Software Reuse	2	2	4
		SE-204 B	OOAD using UML			
<b>Practical /Lab</b>						
5	<b>Lab / Practical</b>	SE-205	Lab-3: Advanced Java	1	1	2
		SE-206	Lab-4: Linux and DBA	1	1	2
6	<b>Open Elective</b>	SE-207A	University recognized MOOC (NPTEL / SWAYAM / others) OR Intra / Inter Departmental OR Intra / Inter School OR	4	0	4
		SE-207 B	Information Technology			
7	<b>Skill based Activity</b>	SE-208	SK-02	1	0	1
	Total credits					<b>25</b>

**Course Code: SE-101**  
**Paper Title: Programming with C++**

**Course Objectives:**

1. To learn the features of an object-oriented programming language viz., data abstraction, data encapsulation, information hiding, inheritance, and dynamic binding of the messages to the methods.
2. To learn and implement inheritance and enhance problem solving skills in C++ with extensive programming projects.

**Course Outcomes:**

1. Describe the procedural and object oriented paradigm with the basic concepts of streams, classes, functions, data and objects.
2. Describe the concept of function overloading, operator overloading, static members, friend functions, friend classes etc.

**Unit I: Introduction to OOPs & C++**

Introduction to OOPs and its basic features, Basic Components of OOPs, Procedure oriented v/s Object oriented programming, Benefits of OOPs, Applications of OOPs Structure of a C++ program, Decision making, Looping constructs, Arrays, Pointers, Functions, Function prototype , Call by value, Call by reference, Call by address, Return by reference , Inline function, Default arguments, Function overloading.

**Unit II: Classes & Objects**

Introduction to Classes & Objects, Access modifiers – private, public, protected Constructors & Destructor, Types of Constructors, Static member data and functions, Friend function and class, Nested classes, Operator Overloading, Operator Overloading through member function and , friend function, Overloading special operators.

**Unit III: Inheritance & Polymorphism**

Introduction to Inheritance, Types of Inheritance, Constructors in Inheritance, Multiple Inheritance, Constructors in Multiple Inheritance Ambiguities in Multiple Inheritance, Introduction to Polymorphism, Virtual Functions & Classes, Pure virtual function and Abstract classes.

#### **Unit IV: Templates, Namespaces, Exception Handling and File Handling**

Function Template and Class Templates, Introduction to STL, Components of STL Creating namespaces, Nesting of namespaces, Exception Handling, Throwing and catching exceptions, Throwing multiple Exceptions, File Handling, Opening and Closing Files, Reading and Writing data, Random access files, File pointers.

#### **Reference Books :-**

1. The C++ Complete Reference, IV Edition, Herbert Schildt, McGraw Hill Publication, 2002, ISBN 0071502394, 9780071502399.

2. Object-Oriented Programming with C++, VI Edition, E Balgurusamy, McGraw Hill, 2013, ISBN 125902993X, 9781259029936.



**Course Code: SE-102**  
**Paper Title: Software Engineering**

**Course Objectives:**

1. To develop software engineering skills and testing plans.
2. To understand system concepts and its application in Software development.

**Course Outcomes:**

After completion of this course students will be able to

1. Learn various methods of software development.
2. Apply various software testing techniques.

**UNIT 1: Software and Software Engineering**

The Evolving Role of Software, Software Characteristics, Categories of Computer Software, The Software Myths, Software Engineering – A layered Technology, The software process, The nature of Software, Legacy Software.

**UNIT 2: Process models**

A generic process model, A Process Framework, The capability Maturity Model Integration (CMMI), Process Patterns ,Process Assessment, The Waterfall Model, Prototyping Model, Spiral Model, Fourth generation techniques, Personal software process, Team software process, Process Assessment and improvement.

**UNIT 3: Requirements Engineering & Design concepts**

Requirements Engineering, Initiating the Requirements Engineering Process, Eliciting Requirements, Negotiating Requirements, Validating Requirements, Developing use cases, Design Process and Design Quality, Design Concepts, The Design Model, Pattern Based Software Design, Web App Design Quality, Design Goals, Web App Engineering Layers, The Web Engineering Process, Web Engineering Best Practices.

**UNIT 4: Software Testing Strategies**

Software Testing fundamentals, A strategic Approach to software Testing, Strategic Issues, Test Strategies for Conventional Software, Validation Testing, System Testing, Debugging, White Box Testing, Black Box Testing, Control Structure Testing, System Testing, Model based Testing, Debugging Process, Debugging Strategies, Correcting the errors.

**Reference Books:**

1. Software Engineering –A Practitioner's approach, Sixth Edition, Roger S. Pressman, McGraw-Hill Higher Education; (1 August 2007),ISBN-10: 0077227808
2. Software Engineering –A Practitioner's approach, Fifth Edition, Roger S. Pressman, McGraw-Hill Higher Education; (1 August 2005)
3. Fundamentals of Software Engineering Second Edition, Rajib Mall, Prentic-Hall India.

**Course Code: SE-103**  
**Paper Title: Operating System Concepts**

**Course Objective:**

1. To introduce concepts of Advanced Operating Systems
2. To learn Multiprocessor and Distributed Operating system
3. To practice on Unix/Linux and Windows as representative examples

**Course Outcome:**

1. Students will be able to understand the working of various types of Operating System
2. Students will be able to write shell script of various operating systems to perform operations

**Unit 1: Overview of Operating System**

Operating System Concepts, Operating System Structure, Process and Thread Management, Memory Management, File System, deadlocks.

**Unit 2: Multimedia Operating Systems**

Introduction to Multimedia, Multimedia Files, Video Encoding, Audio Encoding, The JPEG Standard The MPEG Standard, Multimedia Process Scheduling, Scheduling Homogeneous Processes, General Real-Time Scheduling, Rate Monotonic Scheduling, Earliest Deadline First Scheduling, Placing a File on a Single Disk, Two Alternative File Organization Strategies, Placing Files for Near Video on Demand, Placing Multiple Files on a Single Disk, Placing Files on Multiple Disks, Static Disk Scheduling, Dynamic Disk Scheduling.

**Unit 3: Multiple Processor System**

Multiprocessor, Multiprocessor Hardware, Multiprocessor Operating System Types, Multiprocessor Synchronization, Multiprocessor Scheduling, MULTICOMPUTERS, Multicomputer Hardware, Low-Level Communication Software, User-Level Communication Software, Remote Procedure Call, VIRTUALIZATION, Requirements for Virtualization, Type I Hypervisors, Type 2 Hypervisors, Para virtualization

**Unit 4: Distributed Operating System**

Motivation, Types of Network based Operating Systems, Network Structure, Network Topology, Communication Structure, Communication Protocols, Robustness, Design Issues An Example: Networking, Introduction to distributed file system.

**Unit 5: Case study: Linux**

Linux History, Design Principles, Kernel Modules, Process Management, Scheduling, Memory Management, File Systems, Input and Output, Inter process Communication, Network Structure, Security Summary.

## **Reference Books**

1. Abraham Silberschatz, Peter B. Galvin, G. Gagne, "Operating System Concepts", Sixth Edition, Addison Wesley Publishing Co., 2003
2. Andrew S. Tanenbaum, "Modern Operating Systems", Second Edition, Addison Wesley, 2001
3. Andrew S. Tanenbaum, "Distributed operating system", Pearson education, 2003
4. Mukesh Singhal and N. G. Shivaratri, "Advanced Concepts in Operating Systems", McGraw- Hill, 2000
5. Pradeep K. Sinha, "Distributed operating system-Concepts and design", PHI, 2003

**Course Code: SE-104 A**  
**Paper Title: (Elective) Compute System Security**

**Course Objectives:**

1. Identify the need & importance of policies for the organization.
2. Identify the best mechanism which fulfills most security needs of an organization.

**Course outcome:**

After completion of this course student will be able to:

1. Capable for defining & creating their own security policies.
2. Capable for the real implementation policies according to the policy

**UNIT I: Security Policies, Standard & Guidelines**

Defining policy, Standards & Guidelines, Different Inputs in the development of policies, Different elements of policy, Standard & Guidelines, Policy Examples (Program policy, Information/Resource Classification policy, Standard Access Definition policy, Password Management policy, Internet usage policy, Network Security policy, Remote Access policy, Desktop policy, server Platform policy & Application Security policy), Policy Creation life cycle Information Classification & Access Control Plan: Background, creating Classification, Risk Assessment, Authentication, Authorization, Confidentiality, Authenticity & Integrity, Availability, Non-repudiation, audit/monitoring.

**UNIT II: Applying the Policies to Derive the Requirements**

Threats, External & Internal Security threats, Management security requirements, Defining the security model, Personal Security, Security Awareness and training measures, Change Management, Password Selection & Change Requirements, Business Continuity and Disaster Recovery service.

**UNIT III: Cryptography, Classical Encryption Techniques.**

Basic Introduction to Cryptography, Symmetric & asymmetric encryption, Symmetric cipher model, Substitution techniques (Caesar Cipher, Monoalphabetic cipher, Playfair cipher, Hill Cipher, Polyalphabetic cipher, Transposition techniques, Data Encryption Standard (DES), Advanced Encryption Standard (AES).

#### **UNIT IV: Designing Trusted Operating System & Intruders & Malicious Programs.**

Concept of trusted system, security policies , model of security, Trusted Operating System Design (Trusted system design elements, Security features of general operating system, Security features of trusted O.S Intruders , Intrusion technique, Intrusion detection(Statistical anomaly detection, Rule-based detection & Audit Record. Malicious Programs taxonomy, the nature of viruses, Types of Viruses, Viruses countermeasures, Digital Immune System.

#### **Text Books:**

- 1) Security Architecture Design, Deployment & Operations Christopher M. King, Curtis E. Dalton & T. Ertem Osmanoglu. (TATA McGraw Hill)
- 2) Cryptography & Network Security Principles & Practices (IV Edition) William Stallings
- 3) Security In Computing (Pearson IV Edition) Charles P. Pfleeger & Shari Lawrence Pfleeger.

**Course Code: SE-104 B**

**Paper Title: (Elective) Principles of Programming Language Concept**

**Course Objectives:**

1. Identify the need of programming languages with associated syntax & semantics.
2. Identify the role of programming language and its environment.

**Course Outcome:**

After completion of this course student will be able to:

1. Understanding the different language design issues.
2. Understanding the different language translation issues.

**UNIT I: Language Design Issues**

Need of Programming languages, History of programming languages, Impact of Programming Paradigm, Role of programming language, Programming Environment, Impact of Machine architecture: The Operation of a computer (Computer Hardware, Firmware, translator and virtual architectures, virtual Computers & language implementation.

**UNIT II: Language Translation Issues**

Programming language syntax (General Syntactic criteria, Syntactic elements of language, overall program-subprogram structure, Stages in translation (Analysis of source program & Synthesis of the object program, Formal Translation Models (BNF Grammar, Finite-State Automata, Pushdown automata, General parsing algorithms, **Pascal overview**

**UNIT III: Elementary Data Types**

Properties of types and objects (Data objects, variables, constants) , data types, declaration, type checking and type conversion, assignment & initialization, Scalar Data Types: Numerical data types, Enumeration, Booleans & Characters, Composite Data Types: Character String , pointers & Programmer-constructed data objects, files and input output, **FORTRAN Overview**

**UNIT IV: Storage Management & Distributed Processing**

Elements Requiring Storage, Programmer-and system-controlled storage, Static storage management, heap storage management (LISP overview, fixed & Variable size elements, Distributed Processing: Variations on subprogram control (Exceptions & Exception handler, coroutines, scheduled subprogram), Parallel Programming( Concurrent execution, Guarded Commands, Ada overview, Task & synchronization of task.

**Reference Books:**

- 1) Programming Languages Design & Implementation (Pearson Education 4<sup>th</sup> Edition)  
Terrance W.Pratt, Marvin V. Zelkowitz, T.V. Gopal
- 2) Theory of Computer Science PHI 3<sup>rd</sup> Edition K.L.P. Mishra, N.Chandrasekaran.

**Course Code: SE-105**

**Paper Title: Lab-1 (Programming in C++)**

Practical on C++ Programming At least 15 Logical programs covering all concepts of OOP

**Course Code: SE-106**

**Paper Title: Lab-2 (Based on Elective Subjects )**

Practical on Elective subject SE-104A or SE-104B depending upon choice of students

Code: SE- 107 A	First semester	<b>Open Elective</b>	Credits: 04
<b>Open Elective : University recognized MOOC (NPTEL / SWAYAM / others) OR Intra / Inter Departmental courses</b>			

**OR**

**Course Code: SE-107 B**  
**Paper Title: Research Methodology**

**Course Objectives:**

1. Introduce students to social psychological research methods
2. Understand science and its relationship to social issues

**Course outcome:**

After completion of this course student will be able to:

1. Understand basic concepts of research and its methodologies
2. Able to identify appropriate research domains

**Unit 1: Introduction to Research methodology**

Meaning of Research, Objective of Research, motivation of Research, types of Research, Research Approach, significance of Research, Research & scientific Method Research Processes  
 What is research Problem, Selecting a Problem ,necessity of designing Problem, Techniques involved in defining Problem,

**Unit 2: Research designing and Sampling**

Meaning of research design, Need for Research Design, Features of Good design, Important concept relating to research design, Different Research design, Basic Principles of experimental design, Introduction of sampling, Steps in sampling design, Criteria of Selecting a sampling Procedure, Types of Sampling design, How to select random sample, Random sample from an infinite Universe.

**Unit 3: Scaling Techniques and data Collection**

Measurement in research, Measurement scale, Sources of error in measurement, Test of Sound Measurement, Techniques of developing Measurement Tools, Important Scaling techniques , Methods for collection of Primary data ,Data collection through questionnaires & scheduling ,



Difference Between questionnaires and Scaling, Other Methods of Data Collection, Collection of Secondary data,

#### **Unit 4: Processing Analysis and Testing**

Process Operations, types of analysis, Statistics in research, Measures of Control Techniques , Measure of Dispersion, What is hypothesis, Basic concepts of Hypothesis, Procedure & flow diagram of Hypothesis Testing, Chi-square test bar variance, significance of report writing , Different Steps in writing report, types of report

#### **Reference books**

- 1) Research Methodology Methods & Techniques-By - C.R.Kothari (2nd rev. edition) (New Age Int. Publishers)
- 3) Research Methodology- By- Ratan Khasnabis & saha (University Press)

Code: SE-108	First semester	<b>Skill based Activity</b> <b>SK- 01: PC Assembly and Maintenance</b>	Credits: 01
Scope : Practically understand the PC and surrounding peripherals. The student will assemble / setup and upgrade personal computer systems; install OS and other application software, diagnose and isolate faulty components; optimize system performance and install / connect peripherals.			

**Course Code: SE-201**

**Paper Title: Linux Operating System Concepts and Administration**

**Course Objectives:**

1. This course shall build a platform for students to start their own enterprise
2. For Making Student Job Ready

**Course Outcome:**

After completion of this course students will be able to

1. Understand the Linux OS architecture.
2. Install and use different types of distributions available in market.

**UNIT 1-Introduction to Linux**

Introduction to Linux ,Advantages of Linux ,Distributions of Linux ,Linux Installation Process ,Hardware Requirements of Linux ,Linux Partitioning ,virtual Memory Space(swap) ,GNOME & KDE Desktop , Boot Loader , Login and shutdown

**UNIT 2-Linux Shell & file Structure**

Linux File System , Linux shell and its types , Text Editor , Working with Linux console-text based and virtual based , File name expansion, concept of Pipe, Job:-Background ,Kill & Interruption , Ending Process-PS & KILL , X window system ,Configuration of X window , File system-EXT-2,EXT-3 .

**UNIT 3-Merging and System user Administration**

Linux system user Control:-Root user, Root Password, Root user access , System run Levels: telinit, init tab and shutdown:-Run levels, Run levels in init tab, changing run levels with tel init, run level commands , shutdown , Performance analysis tools and Processes, Merging groups - group management tools , Adding and removing user with useradd, usermod and userdel, File permissions, Managing Password-policy& shadow password, Managing disk quotas, Configuration and managing Print services, Local Printer configuration, Network Printer Configuration .

**UNIT 4-Managing Network Connectivity & security**

TCP/IP Network address -IPV4 & IPV6 , Class of IP address ,Difference between IPV4 and IPV6, Domain name service, Monitoring network using ping, netstat, tcpdump , Ether Ape ,Ether cap, Dynamic host configuration protocol(DHCP), Concept of Comba, Concept and Installation of Apache win server, Public key encryption, integrity check & Digital signature, Wireless Networking.

**Reference books:**

- 1) The Complete Reference of Linux - By Richard Petersen (6th edition)
- 2) Fedora Unleashes - By Bill Ball & David Pits

**Course Code: SE-202**  
**Paper Title: Advanced Java Programming**

**Course Objectives:**

1. To develop background knowledge as well as core expertise AWT, Frames, Applet etc.
2. To understand the dynamic web page creation and provide knowledge for creating Dynamic websites.

**Course outcome:**

After completion of this course students will be able to

1. Impart the knowledge on basics concepts of multithreading programming.
2. Outline the various AWT classes.

**UNIT 1: Introduction to Java and Object Oriented Programming**

Why Java is important for Internet, Java Magic: Byte Code, Java Buzzwords, Simple program of java, Using super keyword, Dynamic method dispatch, Final class and Methods, Packages, Access Protections, Interfaces, Exception Handling Fundamentals, Working with finally clause.

**UNIT 2: Multithreading, Applet and Event Handling**

Multithreading Basics, Creating and Running a Thread, Thread life cycle, Thread Priorities, Thread synchronization, Applet Fundamentals, Applet Architectures, An Applet skeleton, The HTML APPLET tag, Passing parameters to Applet, Event class, Event Types and Listener, Action Event, Mouse Event, Key Event, Windows Event.

**UNIT 3: Introduction to AWT & SWING**

AWT Classes, Windows Fundamentals, Working with Frame window, Working with Graphics, Working with Colors & Fonts, Layout Managers, Swing & Its Features, JApplet, Icons & Labels Button & Label, Text Field & Toggle Buttons, Checkboxes, Radio buttons, Combo Box & Lists, Scroll panes, Trees, Tables, Menu Bars & Menus, Tool Bars, Dialog Boxes, File Dialog, Progress Bar, Choosers

**UNIT 4: Java Beans & JDBC**

Introduction & Advantages of JavaBeans, Application Building Tools, Bean Development Kit, JAR Files, Developing Simple Bean Using the BDK, The Java Bean API, Introduction to JDBC, Types Of JDBC Connectivity, Accessing Relational Database from java Programs, Establishing database Connection.

**Reference Books:**

1. Java Complete Reference by Herbert Schildt Tata McGraw-Hill.Publisher: Sams 2000.
2. Mastering Java2 J2SE1.4 by John Zukouski PBP Publication
3. Java How to Program By H.M Deitel, P.J. Deitel 6th Edition.

**Course Code: SE-203**  
**Paper Title: Database Administration**

**Course Objectives:**

1. To introduce students about the concepts of database administration
2. To teach students about different types of databases.

**Course Outcomes:**

After completion of this course students will be able to

1. Distinguish between data administration and database administration
2. Explain the concept of system performance, backup and recovery

**UNIT 1: Database Architecture**

Overview of database, pfile, spfile, Instance, Table spaces, Data files, Other files, Oracle managed Files, Users, Schemas, Indexes, View, Sequences, Synonyms, Privileges, Roles, Clusters, Hash Clusters, Internal memory structure, SGA, PGA ,Background processes, External structure, Redo logs, Control files, Trace files, Alert logs, Creating database manually, Architectural overview, Standalone hosts, Multiple databases, Networked hosts, Networks of databases, Remote updates, Remote application options, Real application, Clusters, Multiple processors, The parallel query and parallel load options, Client/server databases application, Standby databases.

**UNIT 2: Physical & Logical Database Layouts**

Database file layouts, I/O connections among data files, I/O bottlenecks among all data files, Concurrent I/O among background processes, Defining recoverability and performance goals for the system, Defining the system hardware and mirroring architecture, Database space using overview, Implementation of the storage clause, Locally managed Tablespaces, Dictionary managed Tablespaces, logical structure of a database, Different types of Tablespaces, Changing the Tablespaces size, Allocating segments for temporary segments, Temporary segments in permanent Tablespaces, Changing table space status, changing table space storage settings, Oracle Managed Files (OMFs), Oracle Flexible Architecture (OFA), Different segments types and relationships, Extent usages, Block space utilization.

**UNIT 3: Backup & Recovery**

Types of Logical and Physical backups, Implementations , Integrations of backup procedures, NOARCHIVELOG Mode, ARCHIVELOG Mode, Backup Methods –Closed Database Backup, Open Database Backup, Recovery in NOARCHIVELOG Mode, Recovery in ARCHIVELOG Mode, Recovery manager architecture, Recovery Manager Features, Using Recovery manager & RMAN, Using OEM backup manager, Generating lists and reports.

## **UNIT 4: Database Security & Auditing**

Security capabilities-Account security, Object privileges, System level roles and privileges, Implementing security-operating system security, Create user, Drop user, User profiles, and Password managements, Preventing password reuse, setting password complexity, Using password file for authentication, Auditing, Login audits, Action audits, Object audits, Protecting the audit trail.

### **Reference Books**

1. Oracle 9i DBA Handbook, Eighth Reprint - Kevin Lonely, Marlene Theriault Oracle Press, Tata McGraw Hill Publication ISBN-0- 07-048674-3.
2. OCA Oracle 9i Associate DBA Certification Exam Guide, Sixth Reprint, Jason Couchman, Sudheer N. Marishetti Oracle Press,Tata McGraw Hill Publication,2005 ISBN-0-07-049893-8

**Course Code: SE-204 A**  
**Paper Title: (Elective) Software Reuse**

**Course Objectives:** The purpose of this course is

1. To explain about software reuse, various ways to develop reusable components, metrics and benefits.
2. To provide a solid background knowledge about software Reuse.
3. To educate Metrics used in software reuse.
4. To provide Knowledge about various frameworks

**Course Outcomes:** After completion of this course students will be able to understand all models and processes for Reverse Engineering

**UNIT 1 - INTRODUCTION : SOFTWARE REUSE AND SOFTWARE ENGINEERING**

state the art and the practice - Aspects of software reuse- Software reuse Organizations – Support services – Institutionalizing reuse.

**UNIT 2- DOMAIN ENGINEERING :**

Building Reusable assets – Domain Analysis: Basic concepts – domain scoping – Domain vs application requirements – Domain analysis methods – Domain analysis Tools- Programming paradigms and reusability.

**UNIT 3 - OBJECT ORIENTED DOMAIN ENGINEERING :**

A pragmatic introduction to object orientation: Introduction- The Tenets of object oriented programming. Abstraction and parameterization techniques in object orientation: Abstraction techniques in object oriented modeling – Abstraction techniques in object oriented programming languages – Meta-programming – Design patterns.

**UNIT 4 - FRAMEWORKS AND APPLICATION ENGINEERING :**

Application Frameworks: Framework – Fulfilling the framework contract – Building frameworks – The SWING framework. Architectural frameworks: Architecture – Architecture and reuse – CORBA – Application Engineering – Component storage and retrieval – Reusable asset integration.

**UNIT 5-MANAGERIAL ASPECTS OF SOFTWARE REUSE :**

Software Reuse metrics – Software reuse cost estimation – software reuse return on Investment – Component based software engineering – product-line Engineering – COTS based development.

**Reference Books**

1. Hongji Yang (De Montfort University, UK) and Xiaodong Liu (Edinburgh Napier University, UK), “Software Reuse in the Emerging Cloud Computing Era”, 2012.
2. Hafeedh Mili, Ali Mili, Sherif Yacoub, Edward Addy, “Reuse-Based Software Engineering: Techniques, Organizations, and Control”, John Wiley & Sons, 2002.
3. Carma McClure, “Software Reuse: A Standards-Based Guide”, IEEE, 2001.
4. Wayne C. Lim, “Managing Software Reuse”, Prentice Hall, 2004.
5. Ivar Jacobson, Martin Gres, Patrick Johnson, “Software Reuse”, Pearson Education, 2004.

**Course Code: SE-204 B**

**Paper Title: (Elective ) Object Oriented Analysis & Design using UML**

**Course Objectives:**

1. This course shall build a platform for students to start thinking on OO based Analysis and Design
2. To gain an understanding of the processes that are involved in OOAD

**Course outcome:** After completion of this course students will be able to

1. Understand the OOAD process and components
2. Install and use UML tools for actual development

**Unit-1: Overview of Object Oriented Systems Development:**

Two Orthogonal Views of the Software, Concept of Object Oriented Software, Importance of Object Oriented Software, Object Oriented Future, Object Oriented Systems Development Methodology, Overview of Unified Approach.

**Unit-2: Object Basics:**

An Object Oriented Philosophy, Objects, Object Behavior, Object Oriented Properties, Association and Aggregation.

**Unit-3: Object Oriented Systems Development Life Cycle:**

The Process of Software Development, Developing Good Quality Software, Use Case Driven Approach for Object Oriented Systems Development, Reusability.

**Unit-4: Object Oriented Methodologies:**

Introduction, Types of Object Oriented, Methodologies, Patterns, Unified Approach.

**Unit-5: Unified Modeling Languages (UML):**

Overview of Unified Modeling Language (UML), Static and Dynamic Models, UML Diagrams, UML Class Diagrams, Use-Case Diagrams, UML Dynamic Modeling, Implementation diagrams, Model Management: Package and Model Organization, UML Extensibility, UML Meta-Model.

**Unit-6: Object Oriented Analysis – Identifying Use-Cases:**

Complexity in Object Oriented Analysis, Business Process Modeling and Business Object Analysis, Use-Case Driven Object Oriented Analysis, Use-Case Model, Developing Efficient Documentation.

**Reference Books**

1. Object – Oriented Modeling and Design- J Rumbaugh , M Blaha, PHI Publication.
2. Principles of Object- Oriented Software Development - Anton Eliens , Addison Wesley.
3. Object Oriented System Development - Ali Bahrami McGraw-Hill international Edition
4. UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition), Martin Fowler



**Course Code: SE-205**  
**Paper Title: Lab-3 Advanced Java**

At least 20 Practical's in Advanced Java covering all concepts in theory course

**Course Code: SE-206**  
**Paper Title: Lab- 4 : Linux and DBA**

At least 20 Practical's covering issues like installation, tuning, performance improvement, troubleshooting, database connectivity, database managements, security privileges, migrations, etc

Code: SE- 207 A	First semester	<b>Open Elective</b>	Credits: 04
<b>Open Elective : University recognized MOOC (NPTEL / SWAYAM / others) OR Intra / Inter Departmental courses</b>			

**OR**

**Course Code: SE-207 B**  
**Paper Title: Information Technology**

**Course Objectives:**

1. Introduce students to foundation of Information technology

**Course outcome:** After completion of this course student will be able to:

1. Understand basic concepts in IT and their use in actual working

**Unit 1: Computer Organization, Memory and Storage**

Introduction, Basic Computer Organization, Input Devices, Output Devices, Central Processing Unit, The System Bus Architecture, Memory or Storage Unit

**Unit 2: Information Technology Basics**

Introduction, Need for Information Storage and Processing, Information Technology Components, Role of Information Technology, Information Technology and the Internet

### **Unit 3: Internet and its Tools**

Introduction, Internet Evolution, Basic Internet Terminology, Data over Internet, Modes of Data Transmission, Types of Networks, Types of Topologies, Protocols used in the Internet, Getting Connected to Internet Applications, Internet Applications, Computer Ethics,

### **Unit 4: Emerging Trends in IT**

Introduction, Electronic Commerce (E-Commerce), Electronic Data Interchange (EDI) Smart Cards, Mobile Communication, Internet Protocol TV

### **Unit 5: Computer Programming and Languages**

Introduction, Planning a Computer Program, Steps for Program Development ,Problem Solving Tools, Program Control Structures, Generations of Computer Languages, Program Methodology, Programming Models

### **Reference books**

Fundamentals of Information Technology, Wiley India Editorial Team,  
ISBN: 9788126543557

Code: SE-208	Second semester	<b>Skill based Activity</b> <b>SK-02: Networking Essentials</b>	Credits: 01
<b>Scope :</b> Networking Essentials deals with knowing what is a network, how to install, configure, and troubleshoot a computer network. It includes knowledge of the fundamental building blocks that form a modern network, such as various cables, switches, routers, connectors, LAN-NIC cards and network operating systems. It then provides in-depth coverage of the most important concepts in contemporary networking like connecting computers/ peripherals, servers and clients, Wi-Fi connectivity, etc. Students are expected to have the skills to build a network / LAN from scratch and maintain, upgrade, and troubleshoot an existing network.			