



Estd. 1962
NAAC 'A' Grade

SHIVAJI UNIVERSITY, KOLHAPUR-416 004. MAHARASHTRA

PHONE : EPABX-2609000 website- www.unishivaji.ac.in

FAX 0091-0231-2691533 & 0091-0231-2692333 – BOS - 2609094

शिवाजी विद्यापीठ, कोल्हापूर – 416004.

दुरध्वनी (ईपीएबीएक्स) २६०९०००० (अभ्यास मंडळे विभाग— २६०९०९४)

फॅक्स : ००९१-०२३१-२६९१५३३ व २६९२३३३.e-mail:bos@unishivaji.ac.in

SU/BOS/Science/

No 0 0 1 5 0

Date: 0 7 DEC 2021

To,

The Head/Co-ordinator/Director
All Affiliated (Science) Colleges/Institutions,
Shivaji University, Kolhapur.

Subject : Regarding syllabi of **M. Sc. Computer Science Part -I Sem I to II (Online Mode)** degree programme under the Faculty of Science and Technology.

Sir/Madam,

With reference to the subject mentioned above, I am directed to inform you that the university authorities have accepted and granted approval to the syllabi and Nature of question paper of **M. Sc. Computer Science Part -I Sem I to II (Online Mode)** under the Faculty of Science and Technology.

This syllabi and nature of question paper shall be implemented from the Academic Year 2021-2022 onwards. A soft copy containing the syllabus is attached herewith and it is also available on university website www.unishivaji.ac.in (students Online Syllabus)

You are, therefore, requested to bring this to the notice of all students and teachers concerned.

Thanking you,

Yours faithfully,


Dy Registrar

Copy to:

1	The Dean, Faculty of Science & Technology	7	Appointment Section
2	Director, Board of Examinations and Evaluation	8	P.G.Seminar Section
3	The Chairman, Respective Board of Studies	9	Computer Centre/ I. T.
4	B.Sc. Exam	10	Affiliation Section (U.G.)
5	Eligibility Section	11	Affiliation Section (P.G.)
6	O.E. I Section	12	P.G.Admission Section

Shivaji University, Kolhapur



Accredited By NAAC with 'A++' grade with CGPA 3.52

Syllabus for

Master of Science (M.Sc) Part-I

In

Computer Science

(Online Mode)

(Subject to modifications to be made time to time)

Syllabus to be implemented from 2021-2022

Introduction

Increasing applications of computers in almost all areas of human Endeavour has made almost everyone a computer user, and many of them even computer programmers. Computer Science can be seen on a higher level, as a science of problem solving with reasoning. M.Sc. Computer Science has been growing as an important Programme under faculty of Science and Technology. Computer Science is a discipline that covers both theoretical and practical aspects of emerging technologies. It requires thinking both in abstract terms and in concrete terms. It provides the platform for learning and refinement.

M.Sc. Computer Science in Online Mode will commence at Shivaji University, Kolhapur, Maharashtra, India in this year 2021, to cater to the needs of working professionals, students, and enthusiastic learners who cannot attend traditional classroom teaching learning and wants to upgrade their skills and academics.

Name of the programme: Master of Science in Computer Science(Online Mode)

Duration of the Programme

- The M.Sc. programme will be a full-time two years in **online mode** i.e. 4 semesters. Pattern of examination will be Semester System.

Medium of Instruction

- The medium of Instruction will be English only.

Admission Procedure

- Eligibility: B.Sc. Computer Science (Entire/ optional) / B.Sc. IT/ BCS , B.Sc. Mathematics / Computer Science / Physics / Information Technology / Software Engineering /Statistics / Applied Science - Computer Technology /Applied Science - Information Technology / Applied Science - Computer Systems Maintenance and Networking / Electronics / B.C.A. / B.E.S. or equivalent and a pass in the entrance test conducted by the Shivaji University
- Admission through University Entrance examonly.
- Only entrance marks shall be considered for admissionprocess.
- Reservation of Seats as per rules of Government of Maharashtra.

Fee Structure: -As per the University norms decided by time totime.

Standard of Passing:

1. There would be single head of passing. For university (external) examination and internal evaluation,40% of total marks have to be secured by student per course.

Passing Rules:

1. For admission to M.Sc. Part-II, a candidate must have cleared at least 4 papers of M.Sc. Part I.
2. Students have to complete M.Sc. program within 2+4 years from the date of admission.

Assessment:

The final total assessment of the candidate is made in terms of an internal assessment and an external assessment for each course.

All internal and external assessment will be conducted in online mode.

1. For each theory paper, 20% marks will be based on internal assessment and 80%marksforsemester end examination..
2. Laboratory course assessment will be internal.

Nature of question paper:

Nature of question paper is as follows for University end semester examination

a. Theory Examination:

Nature of Question paper and Scheme of marking for all papers:

Question paper will be divided into two parts on the following format:

Part – A: 25 MCQ's of two marks each. (All questions are compulsory).

Part – B: There will be five questions of 10 marks each, Student has to attempt any three questions.

Section	Marks
A	50
B	30

1. Total marks allotted: 80Marks
2. Time allotted: 150 minutes
3. No negative marking

b. **Laboratory Examination:** Laboratory course assessment will be internal.

Online Pedagogy Transaction Activities:

S. No	Credit value of the course	No. of Weeks	No. of Interactive Sessions		Hours of Study Material		Self-Study hours including Assessment etc.	Total Hours of Study (based on 30 hours per credit)
			Synchronous Online Counseling / Webinars/ Interactive Live Lectures (1 hour per week)	Discussion Forum/ asynchronous Mentoring (2 hours per week)	e-Tutorial in hours	e-Content hours		
1	4 credits	12 Weeks	12 hours	24 hours	20	20	44	120

Requisite attainment by the students for appearing online learning of each course-

- (i) The learner has **minimum participation of 75 % in all** of online programme prior to end semester examination or term end examination.
- (ii) After completion of minimum participation of Online Pedagogy Transaction Activities, learner will be able to approach for the final examination.
- (iii) After successfully completing above mentioned (i) and (ii), then students will be given credits as follows:

Grade Point Table

Range of Marks obtained out of 100 or any fractions	Grade Points
0	0 To 5
1	6 To 10
1.5	11 To 15
2	16 To 20
2.5	21 To 25
3	26 To 30
3.5	31 To 35
4	36 To 40
4.5	41 To 45
5	46 To 50
5.5	51 To 55
6	56 To 60
6.5	61 To 65
7	66 To 70
7.5	71 To 75
8	76 To 80
8.5	81 To 85
9	86 To 90
9.5	91 To 95
10	96 To 100

Grading: A Seven-point grading system as follows:

Grades	CGPA Credit Points
O	8.60 To 10
A+	7.00 To 8.59
A	6.00 To 6.99
B+	5.50 To 5.99
B	4.50 To 5.49
C	4.00 To 4.49
D	0.00 To 3.99

Overall Final Grades:

CGPA Credit Points	Class		Grade
8.60 To 10	Higher Distinction Level	Extraordinary	O
7.00 To 8.59	Distinction Level	Excellent	A+
6.00 To 6.99	First Class	Very Good	A
5.50 To 5.99	Higher Second Class	Good	B+
4.50 To 5.49	Second Class	Satisfactory	B
4.00 To 4.49	Pass	Fair	C
0.00 To 3.99	Fail	Unsatisfactory	D

Programme Educational Objectives(PEO's)

Program Educational Objectives is to prepare Post graduates to:

1. Apply and continuously acquire knowledge, both theoretical and applied, related to core areas of computerscience
2. Demonstrate the ability to work to cater to ever-changing professionalenvironment
3. Workproductivelyascomputerprofessionals(intraditionalcareers,graduateschool, or academia) by demonstrating effective use of oral and written communication, workingcompetentlyby,adheringtoethicalstandardsinthe profession.

Programme Outcomes (POs)

At the end of the Master of Science in Computer Science Programme, graduating students/graduates will be able to:

1. Communicate computer science concepts, designs, and solutions effectively and professionally
2. Applyknowledgeofcomputingtoproduceeffectivedesignsandsolutionsforspecific problems
3. Identify, analyze, and synthesize scholarly literature relating to the field of computer science Use software development tools, software systems, and modern computing platforms.

4. Apply design and development principles in the construction of software systems of varying complexity.

Programme Specific Outcomes (PSOs)

1. Demonstrate understanding of the principles and working of the hardware and software aspects of computer systems.
2. Ability to understand the structure and development methodologies of software systems.
3. Possess professional skills and knowledge of software design process. Familiarity and practical competence with a broad range of programming language and open source platforms.
4. Be acquainted with the contemporary issues, latest trends in technological development and thereby innovate new ideas and solutions to existing problems.

Programme structure

M.Sc. Part I - Semester I

Course Code	Title of the Course	Credits	Teaching Scheme (h/w)		Evaluation Scheme (marks)		
			L	P	CIE	SEE	Total
CC-101	Design and Analysis of Algorithms	4	**Contact sessions will be conducted as per UGC norms for Delivery of Courses in online mode		20	80	100
CC-102	Operating system	4			20	80	100
CC-103	Database Management System	4			20	80	100
CC-104	Cyber Security	4			20	80	100
CCPR-105	Database Lab	4			100	--	100
Total		20			180	320	500

M.Sc. Part I - Semester II

Course Code	Title of the Course	Credits	Teaching Scheme (h/w)		Evaluation Scheme (marks)		
			L	P	CIE	SEE	Total
CC-201	Java Programming	4	**Contact sessions will be conducted as per UGC norms for Delivery of Courses in online mode		20	80	100
CC-202	Python Programming	4			20	80	100
CC-203	Software Engineering	4			20	80	100
CCPR-204	Java Programming Lab	4			100	--	100
CCPR-205	Python Programming Lab	4			100	--	100
Total		20			260	240	500

Total Marks for M.Sc.-I : 1000
Total Credits for M.Sc. -I (Semester I & II) : 40
CC-Core Course
CCPR-Core Course Practical
CIE – Continuous Internal Examination
SEE – Semester End Examination

**UGC norms for Delivery of Courses in online mode

M.Sc. Part II - Semester III

Course Code	Title of the Course	Credits	Teaching Scheme (h/w)		Evaluation Scheme (Marks)		
			L	P	CIE	SEE	Total
CC -301	Computer Networks	4	**Contact sessions will be conducted as per UGC norms for Delivery of Courses in online mode		20	80	100
CC-302	Advanced Java Programming	4			20	80	100
CC-303	Web Programming	4			20	80	100
CE-304	Elective-I: CE-304.1: Image Processing CE-304.2 Cloud Computing	4			20	80	100
CCPR-305	Advanced Java Programming Lab	4			100	--	100
CCPR-306	Web Programming Lab	4			100	--	100
Total		24			280	320	600

M.Sc. Part II - Semester IV

Course Code	Title of the Course	Credits	Teaching Scheme (h/w)		Evaluation Scheme (Marks)		
			L	P	CIE	SEE	Total
CC -401	Artificial Intelligence	4	**Contact sessions will be conducted as per UGC norms for Delivery of Courses in online mode		20	80	100
CC-402	Advanced Web Technology	4			20	80	100
CC-403	Machine Learning Using Python	4			20	80	100
CE-404	Elective-II: CE-404.1: Advanced Computer Architectures CE-404.2: Network Security	4			20	80	100
CCPR-405	Advanced Web Technology Lab	4			100	--	100
CCPR-406	Machine Learning Lab	4			100	--	100
Total		24			280	320	600

Total Marks for M.Sc.-II : 1400	
Total Credits for M.Sc.-II (Semester III & IV):48	
CC-Core Course	
CE-Core Elective	
CIE – Continuous Internal Examination	
SEE – Semester End Examination	
CCPR – Core Course Practical.	Total Credits for M.Sc. Course : 88
	Total Marks for M.Sc. Course : 2200

**UGC norms for Delivery of Courses in online mode

M.Sc.-I Semester-I(Computer Science)(Online Mode) CC-101:Design and Analysis of Algorithms

Course outcomes:

1. Analyze the asymptotic performance of algorithms.
2. Compare algorithms based on time & space complexity.
3. Employ graphs to model real-life problems, when appropriate. Develop algorithms that employ graph computations as key components, and analyze them.
4. Mapping of data structures like Stack, Queue and Linked List to real-life problems.
5. Be familiar with advanced data structures such as balanced search trees, hash tables, B-trees.
6. Understand Divide & Conquer approach and Greedy algorithm for algorithm design.

Unit I

Algorithm Analysis: Introduction to algorithms, analyzing and designing algorithms, Growth functions, asymptotic notations, solving recurrences: Substitution method, recursion tree method, master method. **Hashing:** Hashing, Direct address tables, Hash tables, Hash functions, collision resolution techniques.

Unit II

Linear Data Structures: Arrays : Representation, Multidimensional Arrays, sparse matrices, sparse matrix representation. **Linked List:** Processing linked list, circularly linked list, doubly linked list, multilinked lists. **Stacks :** Processing the stacks, Linked list implementation, Application of Stacks for expression solving, Non recursive implementation of recursive algorithm. **Queues :** Processing the queues, Linked list implementation, Deques, Priority queues and their applications.

Unit III

Non-linear data structure: Trees : Representation of hierarchical relationships, General Trees, Binary trees, Binary search trees, linked list implementation, traversal algorithms, height balanced trees, Heap tree, B-tree indexing. **Graph:** Graph representations, Breadth first and Depth first search, Topological sort, Single source Shortest path, Minimum Spanning tree, applications of graph.

Unit IV

Algorithm design approaches: Greedy algorithm: Characteristics, Problem solving using Greedy Algorithm - Job Scheduling Problem, Huffman code

Divide and conquer: Characteristics, Problem solving using Divide and conquer approach- Binary Search, Quick sort, Merge sort

References:

1. Introduction to algorithms, Third Edition. by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, PHI
2. Fundamentals of Computer Algorithms, Second edition. By Ellis Horowitz, Sartaj Sahani, Sanguthevar Rajasekaran, University Press.
3. Data structures and algorithm analysis in C, Second edition. By Mark Allen Weiss

4. Fundamental algorithms by Donald E. Knuth, Pearson Education.

M.Sc.-I Semester-I (Computer Science) (Online Mode) **CC-102: Operating Systems**

Course Outcomes:

1. To understand the basic concepts and functions of operating systems.
2. To understand Processes and Threads
3. To analyze Scheduling algorithms.
4. To understand the concept of Deadlocks.
5. To analyze various memory management schemes.
6. To understand I/O management and File systems.
7. To be familiar with the basics of Distributed Operating System

Unit I

Introduction: Operating system definition, Functions of Operating System, Logical View, System View, Types of operating System, System Calls, System Programs, Interrupt Concept, Concept of Virtual Machine, **Processes:** Process Concept, Thread Concept, Difference between Process and Thread, Process Control Block, Process operations, Inter-process Communication.

Unit II

CPU Scheduling: Scheduling Concept, Scheduling Criteria, Scheduling algorithms, Scheduling Evaluation, Simulation Concept, Numerical Exercises Based on CPU Scheduling Algorithms. **Process Synchronization:** Synchronization concept, Need for Synchronization, Critical Section Problem, Semaphore, Monitor. **Deadlock:** Deadlock concepts, Necessary Conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance, Bankers Algorithm, Deadlock Detection, Deadlock Recovery.

Unit III

Memory Management: Concept, Memory Management Techniques, Contiguous & Non Contiguous allocation, Relocation, Compaction, Logical & Physical Memory, Conversion of Logical to Physical address, Paging, Segmentation, Segment with paging, Virtual Memory Concept, Demand paging, Page fault, Need for Page Replacement, Page Replacement algorithms, Thrashing,.

Unit IV

Disk Management: Disk Structure, Disk Scheduling algorithm, Disk management, Swap Space concept and Management, RAID structure, Disk performance issues, **Distributed Operating System:** Difference between Distributed & Centralized OS, Advantages of Distributed OS, Types of Distributed OS.

References:

1. Operating System Silberschatz, Galvin , Gagne, Wiley publication
2. Operating System Concepts and Design , Milan Milenkovic, MGH
3. Distributed Operating System P.K. Sinha, PHI
4. Operating system Achyut Godbole
5. Operating System In Depth Doeppner Wiley India
6. Operating System Rohit Khurana Vikas pub

M.Sc.-I Semester-I(Computer Science) (Online Mode)

CC-103:Database Management System

Course Outcomes:

1. Demonstrate an understanding of the relational data model.
2. Transform an information model into a relational database schema and use a data definition language and/or utilities to implement the schema using a DBMS.
3. Formulate, using SQL, solutions to a broad range of query and data update problems.
4. Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.
5. Use an SQL interface of a multi-user relational DBMS package to create, secure, populate, maintain, and query a database.
6. Use PL/SQL for handling data in a database as per the user's requirement using programming features.
7. Define various cursors and its implementation along with procedure and functions.

Unit I

Introduction to DBMS: Database Concept, Characteristics and architecture of DBMS, Database users, 3-tier architecture of DBMS-its advantages over 2-tier, Views of data-Schemas and instances, Data independence. Data models: Conventional data models, NDM & HDM, Physical Data Organization-Hashed files, Indexed files, B-trees

Unit II

Database Design and E-R Model: Entities, Attributes, Relationships, Representation of entities, relationship set, Generalization, aggregation Structure of relational Database and different types of keys, E-R diagrams, E-R design Issues in Relational database design, Functional dependencies. Normalization and Database Recovery systems: Codd's rules, Normalization, Database Recovery System-Failure classification, storage structure, recovery and atomicity, log-based recovery. Role of DBA.

Unit III

Introduction to RDBMS: History, Generations and characteristics, difference between DBMS & RDBMS. Data constraint-primary key, foreign key, unique key, null, not null, default key etc. SQL: Introduction to SQL, Features of SQL, Basic data types, SQL statements/commands, Set operations in SQL, order by and group by clause, like between, in, like, create index, view and join command Nested queries, GRANT and REVOKE, Commit, Rollback, Savepoint. Join concept: Simple, Equi, non-equi, Self, Outer join. View- Introduction, Create, Update, Drop, Index. SQL functions: MAX, MIN, SORT, COUNT, AVERAGE, Numeric, String, Date Functions, Type conversion functions.

Unit- IV

Introduction to PL/SQL: Introduction, Difference between SQL and PL/SQL, Block definition structure and Data types, Block Functions - %Type, %Row Type, Control statements, Looping statements and sequential statement, Exception handling. Simple PL/SQL blocks.

Cursor management: meaning, types and importance, implicit and explicit cursor management using simple example. Trigger: meaning importance and types of trigger, examples using trigger Procedures- Definition, creating procedures, passing parameters. Function- Definition, syntax and calling methods, passing parameters.

Reference Books

1. Introduction to database systems C.J.Date Pearson Education 8th
2. Database system concept Korth, Silberschatz and Sudarshan MGH 5th
3. Fundamentals of Database Systems Elmasri Navathe Pearson Education 5th
4. SQL/PLSQL For Oracle 11G BlackBook Dr. Deshpande Wiley Dreamtech 2012
5. ORACLE PL/SQL Programming Scott Ulman TMH 9th
6. SQL, PL/SQL the programming language of Oracle Ivan Bayross BPB 4th
7. Advance Database Management System Chakrabharat/ Dasgupta Wiley Dreamtech
8. Understanding SQL Martin Gruber BPB 2nd
9. SQL Scott Ulman TMH 4th

M.Sc.-I Semester-I(Computer Science) (Online Mode)
CC-104: Cyber Security

Course Outcomes:

1. Realize the need for Cyber Security
2. Understand the fundamentals of Cyber crimes and Cyber security
3. Understand the need for Security in day to day communications
4. Understand the vulnerabilities in the Network and Computer System
5. Understand the cyber law and Cyber Forensics

Unit I

Introduction to Cyber Security: Overview of Cyber Security, Internet Governance – Challenges and Constraints, Cyber Threats:- Cyber Warfare-Cyber Crime-Cyber terrorism-Cyber Espionage, Need for a Comprehensive Cyber Security Policy, Need for a Nodal Authority, Need for an International convention on Cyberspace. Cyber Security Vulnerabilities and Cyber Security Safeguards :Cyber Security Vulnerabilities-Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards- Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.

Unit II

Securing Web Application, Services and Servers: Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges. Intrusion Detection and Prevention: Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, System Integrity Validation.

Unit III

Cryptography and Network Security: Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security Protocols: - security at the Application Layer- PGP and S/MIME, Security at Transport Layer- SSL and TLS, Security at Network Layer-IPSec.

Unit IV

Cyberspace and the Law: Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013. Cyber Forensics: Introduction to Cyber Forensics, Handling Preliminary Investigations, Controlling an Investigation, Conducting disk-based analysis, Investigating Information-hiding, Scrutinizing E-mail, Validating E-mail header information, Tracing Internet access, Tracing memory in real-time.

References:

1. Preston Gralla, How Personal and Internet Security Work, Que Publications
2. Alfred Basta and Wolf Halton, Computer Security Concepts, Issues and Implementation, Cengage Learning
3. Digital Defense: A Cybersecurity Primer by Joseph Pelton , Indu B. Singh
4. Cryptography and Network Security: Principles and Practice by William Stallings
5. Computer and Information Security Handbook by John R. Vacca .
6. Cyberlaw: The Law of the Internet and Information Technology by Brian Craig .
7. Cyber Warfare: Techniques, Tactics and Tools for Security Practitioners 2nd Edition by Jason Andress (Author), Steve Winterfeld (Author)

M.Sc.-I Semester-I (Computer Science) (Online Mode)
CCPR-105: Database Lab

Lab assignments based on Database Management System course.

M.Sc.-I Semester-II (Computer Science) (Online Mode)
CC-201: Java Programming

Course Outcomes:

1. To become familiar with the features of Java Language.
2. To become comfortable with concepts such as Classes, Objects, Inheritance, Polymorphism and Interfaces.
3. To understand Exception handling
4. To design applications using threads.
5. To get familiar with Networking classes
6. To design GUI applications using applets.

Unit I

OOPS in JAVA: Objects and classes, Inheritance, Interfaces, inner classes, Packages. Introduction to Java Utility classes and collection classes - Date, DateFormat and Gregorian Calendar classes. Using ListInterface, ListIterator and LinkedList classes. Set, Iterator, SortedSet, Map interfaces. HashSet class. Using Vector class, stacks, queues, HashTable. Generating random numbers, Property class.

Unit II

Exception Handling and I/O Streams: Exception Handling: Exception class hierarchy, Exception Vs Error, try, catch, throw, throws, finally, checked Vs unchecked exceptions, creating custom exception classes. Significance of streams, various types of Input & Output streams, accessing the file through streams, object serialization. Random Access File.

Unit III

Multi Threading and Networking in Java: Difference between multi tasking and multi threading, Need for multi threading, thread states and priorities, suspending and resuming threads, synchronization between threads. Inter thread communication and dead locks Client and Server programming, Connection oriented and connectionless architectures, Socket, IP address classes. InetAddress, URL and URLConnection classes.

Unit IV

Applets and Event Handling in Java: Applet and its life cycle, passing parameters to applets, font, color, image classes. ImageObserver. Image processing using PixelGrabber and MemoryImageSource classes. Difference between AWT and Swing. Light weight and heavy weight components. Pluggable Look and Feel. Swing package and its components., layout managers, various components for GUI. Delegation Event Model, different types of events, event handlers, and adapter classes.

References:

1. Java Complete Reference by Patric Norton
2. Java 8 Programming Black Book
3. Core Java Vol. I (Addison- Wesley) Sun Press ISBN – 981-405-861-0
4. Core Java Vol. II (Addison- Wesley) Sun Press ISBN – 981-4058-50-5
5. Java in a Nutshell, By Benjamin J Evans, David Flanagan, O'Reilly Media
6. Thinking in Java, Bruce Eckel., Addison – Wesley, ISBN: 9814035750
7. Java 2 Programming Black Book by Steven Holzner, Dream Tech Publication
8. A Programmer's Guide to Java SCJP Certification: A Comprehensive Primer By Khalid Azim Mughal, Rolf Rasmussen
9. Inside Java 2 Virtual Machine by Venner Bill, Mcgraw Hill Education
10. Learning Java by Jonathan Knudsen, Patrick Niemeyer, O'Reilly Media

M.Sc.-I Semester-II (Computer Science) (Online Mode)
CC-202: Python Programming

Course Outcomes:

1. Understand principles of Python
2. Understand object-oriented programming
3. Demonstrate file handling techniques
4. Understand how Python can be used for application development
5. Design real-life problems and think creatively about solutions of them
6. Apply a solution clearly and accurately in a program using Python

Unit I

Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, multiple line statements, designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard

Unit II

A Boolean Type, Choosing Statements to Execute, Nested If Statements, Remembering the Results of a Boolean Expression Evaluation, A Modular Approach to Program Organization, Importing Modules, Defining Modules, Testing Code Semiautomatically Grouping Functions Using Methods: Modules, Classes, and Methods, Calling Methods the Object-Oriented Way, Exploring String Methods, Underscores.

Unit III

Storing Collections of Data Using Lists: Storing and Accessing Data in Lists, Modifying Lists, Operations on Lists, Slicing Lists, Aliasing, List Methods, Working with a List of Lists. Repeating Code Using Loops: Processing Items in a List, Processing Characters in Strings, Looping Over a Range of Numbers, Processing Lists Using Indices, Nesting Loops in Loops, Looping Until a Condition is Reached, Repetition Based on User Input, Controlling Loops Using Break and Continue Reading and Writing Files: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files, Writing Algorithms that use the File-Reading Techniques, Multiline Records

Unit IV

Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary
Creating Graphical User Interface: Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style Widgets, Object-Oriented GUIs, Regular Expressions
Databases: Overview, Creating and Populating, Retrieving Data, Updating and Deleting, Using NULL for Missing Data, Using Joins to Combine Tables, Keys and Constraints.

References:

1. Practical Programming: An Introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo,
2. Python for Informatics: Exploring Information, Charles Severance
3. Learning Python, Fourth Edition, Mark Lutz, O'Reilly publication
4. Introduction to Python for Computational Science and Engineering (A beginner's guide)
5. John V Gutttag. "Introduction to Computation and Programming Using Python",
6. R. Nageswara Rao, "Core Python Programming", Dreamtech Edition by Jason Andress (Author), Steve Winterfeld (Author)

Course Outcomes:

1. Students will get foundation of software engineering, various process models and can apply the new models in development process.
2. Students will have effective communication and interaction skills for requirement engineering tasks.
3. Students can apply design principles for various types of software and designing object oriented software using UML tools.
4. Students can implement testing strategies thoroughly using testing tools.
5. Students will understand the need of lifelong learning and adapt to new software engineering concepts.

Unit I

Introduction to Software Engineering: Software definition, characteristics, software application domains, unique nature of web apps, seven principles of software engineering, software development process, Waterfall Model, prototyping, spiral model, Concurrent Models, The Formal Methods Model, 12 Principles of Agility, Extreme Programming (XP), Scrum process flow, Selection of Software Process models.

Unit II

Requirements Engineering and Design Concepts: Seven tasks of requirement engineering, Eliciting Requirements, Types of requirement, fundamental problem in defining requirements, SRS template. Translating the requirement model into the design model, software design concepts abstraction, architecture, pattern, separation of concerns, modularity, information hiding, functional independence-cohesion, coupling, refinement, aspects, refactoring.

Unit III

Introduction to UML: The design model: Developing use cases, Relationships, class diagrams, associations, generalizations, object diagram, Dynamic modeling – State diagrams, Sequence diagrams, Collaboration diagrams, Activity diagrams etc. Logical and physical architecture – Component diagram, Deployment diagrams etc. Case Studies: Courseware management system, ATM, Airline reservation System

Unit IV

Introduction to Testing: A Strategic Approach to Software Testing, Strategic Issues, Unit testing, Integration testing, Validation Testing, System Testing, Black box testing and white box testing, The Art of Debugging. Testing web applications- testing strategy,

testing process, content testing, user interface testing, navigation testing, configuration testing, security testing, performance testing. Study of software testing tool.

References:

1. Software Engineering by Roger Pressman. 7th edition.
2. Software Engineering for students: A Programming Approach by Douglas Bell, Pearson publication.
3. Software Engineering Sommerville 8th edition.
4. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson AddisonWesley 2005.
5. UML A Beginners Guide Jason T. Roff McGraw Hill Professional.
6. Learning UML 2. 0 Kim Hamilton, Russ Miles O'Reilly Media 2006.
7. Software Quality Engineering by Jeff Tian.
8. Software Testing And Quality Assurance Theory And Practice By Kshirasagar Naik, Priyadarshi Tripathy.
9. The art of software testing by GJ Myers, Wiley
10. Software Testing: Principles and Practices by Srinivasan D and Gopalswamy R, PearsonEd, 2006
11. Software Testing Foundations, Andreas Spillner, Tilo Linz, Hans Schaefer, Shoff Publishers and Distributors.

M.Sc.-I Semester-II (Computer Science) (Online Mode)
CCPR-204:Java Programming Lab

Lab assignments based on Java Programmingcourse.

M.Sc.-I Semester-II (Computer Science) (Online Mode)
CCPR-205:Python Programming Lab

Lab assignments based on Python Programmingcourse.
