॥ सा विद्या या विमुक्तये ॥

स्वामी रामानंद तीर्थ मराठवाडा विद्यापीठ, नांदेड

'ज्ञानतीर्थ', विष्णुपुरी, नांदेड - ४३१ ६०६ (महाराष्ट्र राज्य) भारत

SWAMI RAMANAND TEERTH MARATHWADA UNIVERSITY, NANDED

'Dnyanteerth', Vishnupuri, Nanded - 431 606 (Maharashtra State) INDIA

स्वामी रामानंद तीर्थ मराठवाडा विद्यापीठ, नांदेड

Established on 17th September, 1994, Recognized By the UGC U/s 2(f) and 12(B), NAAC Re-accredited with 'B++' grade

Phone: (02462)215541 **Academic-1 (BOS) Section** **website: srtmun.ac.in**

E-mail: bos@srtmun.ac.in

शैक्षणिक वर्ष २०२४–२५ पासून राष्ट्रीय शैक्षणिक धोरणानुसार लागू केलेल्या विज्ञान व तंत्रज्ञान विद्याशाखेतील पदव्यूत्तर स्तरावरील द्वितीय वर्षाच्या सुधारित (दुरूस्ती) अभ्यासक्रमा बाबत....

प रि प त्र क

संदर्भ:– १. जा.क्र.शै–१/एनईपी/विवत्रंविपदवी/२०२४–२५/११३ दिनांक १३/०६/२०२४

या परिपत्रकान्वये सर्व संबंधितांना कळविण्यात येते की, संदर्भीय परिपत्रकान्वये दिनांक १५ मे २०२४ रोजी संपन्न झालेल्या मा. विद्यापरिषदेच्या बैठकीतील विषय क्र. १५/५९–२०२४ अन्वये मान्यता दिल्यानुसार विज्ञान व तंत्रज्ञान विद्याशाखे अंतर्गत राष्ट्रीय शैक्षणिक धोरणानुसार पदव्युत्तर द्वितीय वर्षाचे अभ्यासक्रम शैक्षणिक वर्ष २०२४–२५ पासून लागू करण्यात आलेले आहेत. तथापी वरील संदर्भीय परिपत्रका अन्वये प्रकाशित केलेल्या अभ्यासक्रमामध्ये अभ्यासमंडळानी किरकोळ दुरूस्ती करून अभ्यासक्रम सादर केला असून मा. अधिष्ठाता, विज्ञान व तंत्रज्ञान विद्याशाखा यांच्या मान्यतेने दुरूस्ती केलेले खालील अभ्यासक्रम लागू करण्यात येत आहेत.

01	M. Sc. II year Computer Science (Campus)
02	M. Sc. II year Computer Application (Campus)

सदरील परिपत्रक व अभ्यासक्रम प्रस्तुत विद्यापीठाच्या **www.srtmun.ac.in** या संकेतस्थळावर उपलब्ध आहेत. तरी सदरील बाब ही सर्व संबंधितांच्या निदर्शनास आणून द्यावी, ही विनंती.

डॉ. सरिता लोसरवार
सहाय्यक.कुलसचिव
शैक्षणिक अभ्यासमंडळ विभाग

'ज्ञानतीर्थ' परिसर,

विष्णुपुरी, नांदेड – ४३१ ६०६.
जा.क्र.:शैक्षणिक–१/परिपत्रक/एनईपीपीजीदुरूस्ती/S&T/
२०२४–२५/271
दिनांक : १४.०९.२०२४

प्रत माहिती व पुढील कार्यवाहीस्तव.

०१. मा. अधिष्ठाता, विज्ञान व तंत्रज्ञान विद्याशाखा, प्रस्तुत विद्यापीठ.
०२. मा. संचालक, परीक्षा व मूल्यमापन मंडळ यांचे कार्यालय, प्रस्तुत विद्यापीठ.
०३. मा. संचालक, सर्व संबधित संकुले, प्रस्तुत विद्यापीठ.
०४. मा. प्राचार्य, सर्व संबधित महाविद्यालये, प्रस्तुत विद्यापीठ.
०५.सिस्टम एक्सपर्ट, शैक्षणिक विभाग, प्रस्तुत विद्यापीठ. यानां देवून कळविण्यात येते की, सदरील परिपत्रक विद्यापीठाच्या संकेतस्थळावर प्रसिध्द करण्यात यावे.

Swami Ramanand Teerth Marathwada University, Nanded

(NAAC Re-accredited with 'B++' Grade)

Syllabus of Second Year

M.Sc. (Computer Science)

(Campus and Sub Centre)*

(2 years) (Common NEP-2020 pattern)

To be introduced from AY 2024-2025

*(BoS deserves the rights for minor corrections, typographical errors in this syllabus with due approval of Administrations)

PEO, PO and CO Mappings

1. **Program Name** : M.Sc.(CS) Common for Campus and Sub Centre

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 computational science 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

   1. **PO1**: Learn, understand and use latest tools in computational world so as to function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary backgrounds in actual software development work
   2. **PO2:** Design component, or processes or programs to meet the needs within realistic constraints.
   3. **PO3:**.Actual hands on technology to understand its working and knowledge of contemporary issues and emerging developments in computing profession
   4. **PO4:** Recognize the need for implementation on applied and interdisciplinary domains / lifelong learning
   5. **PO5:**. Utilize the techniques, skills and modern tools, for actual software development process
   6. **PO6:** 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	All core courses
PEO II	Successful Career	PO1,PO5,PO6	All discipline specific electives courses
PEO III	Hands on Technology and Professional experience	PO2,PO3	All Lab courses
PEO IV	Interdisciplinary and Life Long Learning	PO4,PO5,PO6	All open electives and discipline specific electives

The detailed syllabus is as below,

THIRD SEMESTER

Sr#	Course category	Course Code	Course Title	Total credits
1	**Core Subjects Theory**	SCCSC-501	Advanced Computer Networks	04
2		SCCSC-502	Compiler Designing	04
3		SCCSC-503	Software Engineering	04
4	**Lab / Practical**	SCCSCP-501	Lab 7: CN Lab	01
5		SCCSCP-502	Lab 8: CD Lab	01
Elective Subjects : Choose any one Theory (corresponding elective lab)				
6	**Theory**	SCCSE- 501-		03
		A	Advanced Python Programing	
		B	Network and Linux Administration	
		C	DBMS Programing	
		D	Theory of Computations	
		E	Internet of Things	
		F	Subject relevant MOOC (NPTEL / SWAYAM / RUSA sponsored Future Oriented Courses / other recognized courses[$$] ($$ - with prior consultation with the Director of the School in order to determine relevancy, credit transfer policy will be adopted)	
		G	Visual Programing Tools	
7	**Lab / Practical**	SCCSEP-501	Lab-9: Elective Lab	01
8	**Research Project**	SCCSR-551	Research Project	04

FOURTH SEMESTER

Sr. No	Course category	Course Code	Course Title	Total credits
1	**Core Subjects**	SCCSC-551	AI and Machine Learning	04
2	**Theory**	SCCSC-552	Introduction to Data Science	04
3	**Lab / Practical**	SCCSCP-551	Lab 10: AI-ML Lab	01
4		SCCSCP-552	Lab 11: DS Lab	01
Elective Subjects : Choose any one Theory (corresponding elective lab)				
5	**Theory**	SCCSE-551-	Introduction to Cyber Security	03
		A	Introduction to Cyber Security	
		B	Database Administration	
		C	Software Quality Assurance and Testing	
		D	Principles of User Interface Design	
		E	Cloud Computing Tools and Techniques	
		F	Distributed Computing	
		G	Natural Language Processing	
6	Lab / Practical	SCCSEP-551	Lab-12: Elective Lab	01
7	Publication Ethics	SVECP-551	Publication Ethics	02
8	Research Project	SCCSR-552	Research Project	06

M. Sc. CS Second Year, Semester III and IV *(Level 6.5) : Teaching Scheme*

Course	Course Code	Course Name	Credits Assigned per course			Teaching Scheme (Hrs/ week) per course	
			Theory	Practical	Total	Theory	Practical
Major	SCCSC-501 to SCCSC-503 and SCCSC-551 to SCCSC-552	All Core Course	04	--	04	04	--
Elective	SCCSE-501 and SCCSE-551	All Elective Courses	03	--	03	03	--
Major Practical	SCCSCP-501 to SCCSCP-502 & SCCSCP-551 to SCCSCP-552	All Core labs	--	01	01	--	02
Elective Practical	SCCSEP-501 and SCCSEP-551	Elective lab	--	01	01	--	02
Research Project	SCCSR-551	Research Project	--	04	--	--	04 / 08#
Research Project	SCCSR-552	Research Project	--	06	--	--	06 / 12#
Publication Ethics	SVECP-551	Publication Ethics	02	--		02	--
Total Credits per semester			--	--	**22**	--	--
Total credits per year			--	--	**44**	--	--

 # 01 Credit Lab has 02 hours practical duration so depending upon university policy, the number of hours per week shall be counted

M. Sc. CS Second Year , Semester III and IV (Level 6.5) :Examination Scheme

Course Code (2)	Course Name (3)	Theory				Practical		Total Col (6+7) / Col (8+9) (10)
		Continuous Assessment (CA)			ESA			
		Test I (4)	Test II (5)	Avg of (T1+T2)/2 (6)	Total (7)	CA (8)	ESA (9)	
SCCSC-501 to SCCSC-503 and SCCSC-551 to SCCSC-552	All core courses	20	20	`20	80	--	--	100
SCCSE-501 and SCCSE-551	All elective courses	15	15	15	60	--	--	75
SCCSCP-501 to SCCSCP-502 & SCCSCP-551 to SCCSCP-552	All Core Labs	--	--	--	--	05	20	25
SCCSEP-501 and SCCSEP-551	All Elective labs	--	--	--	--	05	20	25
*SVECP-551	Publication Ethics	As per University Policy						
*SCCSR-551	Research Project	As per University Policy						
*SCCSR-552	Research Project	As per University Policy						

***Note : Teaching scheme and Examination scheme for these courses will be elaborated later**

Guidelines for Course Assessment: M.Sc. CS

A. **Continuous Assessment (CA) (20% of the Maximum Marks):** This will form 20% of the Maximum Marks and will be carried out throughout the semester. It may be done by conducting **Two Tests** (Test I on 40% curriculum) and **Test II** (remaining 40% syllabus). Average of the marks scored by a student in these two tests of the theory paper will make his **CA** score (col. 6).

B. **End Semester Assessment (80% of the Maximum Marks):** ***(For illustration we have considered a paper of 04 credits, 100 marks and need to be modified depending upon credits of an individual paper)***
    1. **ESA Question paper will consists of 6 questions, each of 20 marks.**
    2. **Students are required to solve a total of 4 Questions.**
    3. **Question No.1 will be compulsory and shall be based on entire syllabus.**
    4. Students need to solve **ANY THREE** of the remaining Five Questions (Q.2 to Q.6) and shall be based on entire syllabus.

C. **Question paper of campus and affiliated colleges shall be different**

Note: **Number of lectures required to cover syllabus of a course depends on the number of credits assigned to a particular course. One credit of theory corresponds to 15 Hours lecturing and for practical course one credit corresponds to 30 Hours. For example, for a course of two credits 30 lectures of one hour duration are assigned, while that for a three credit course 45 lectures.**

%%%%%

Course Code:	SCCSC-501	Advanced Computer Networks	Credits: 4

Course Objectives:

To understand the basic concepts of computer network and firm foundation for understanding how data communication occurring using computer network. It is based around the OSI Reference Model which deals with the major issues and related protocol studies in the various layers (Physical, Data Link, Network, Transport, Session, Presentation and Application) of the model.

Course Outcome:

1. Analyse the requirements for a given organizational structure and select the most appropriate networking architecture and technologies;
2. Specify and identify deficiencies in existing protocols, and then go onto formulate new and better protocols;
3. Analyse, specify and design the topological and routing strategies for an IP based networking infrastructure
4. To have a working knowledge of datagram and internet socket programming

Unit-1:	Foundations of Computer Network	15 Lectures

- Introduction to Computer Networks: Definition, History, and Evolution
- Network Models: OSI and TCP/IP Models
- Physical Layer: Transmission Media, Signal Encoding Techniques
- Data Link Layer: Error Detection and Correction, MAC Protocols (Ethernet, Wi-Fi)
- Network Layer: IP Addressing, Subnetting, Routing Protocols (RIP, OSPF, BGP)
- Transport Layer: TCP/UDP, Flow Control, Congestion Control
- Application Layer: DNS, HTTP, SMTP, FTP
- Network Devices: Routers, Switches, Gateways, Hubs

Unit-2:	Advanced Networking Concepts	15 Lectures

- Network Design and Architecture: LAN, WAN, MAN, PAN
- Wireless Networks: Wi-Fi, Bluetooth, Zigbee, LTE, 5G
- Network Security: Firewalls, VPNs, IDS/IPS
- Cryptographic Techniques: Symmetric and Asymmetric Encryption, PKI
- Software-Defined Networking (SDN): Architecture, OpenFlow, Network Virtualization
- Network Function Virtualization (NFV): Concepts, Benefits, Use Cases
- Internet of Things (IoT) Networks: Protocols, Architecture, Security Challenges
- Edge and Fog Computing: Concepts, Architectures, Applications

Unit-3:	Emerging Technologies and Trends	15 Lectures

- 5G and Beyond: Architecture, Key Technologies, Use Cases
- Next-Generation Wi-Fi: Wi-Fi 6, Wi-Fi 6E, Wi-Fi 7
- Quantum Networking: Concepts, Quantum Key Distribution, Challenges
- Blockchain in Networking: Use Cases, Benefits, Challenges
- Cybersecurity Trends: Zero Trust Architecture, SASE
- Cloud Networking: Multi-Cloud Strategies, Cloud Security

Unit-4:	Practical Applications and Case Studies	15 Lectures

- Network Design and Implementation: Case Studies of Enterprise Networks
- Performance Analysis and Troubleshooting: Tools and Techniques
- Smart City Networking: IoT Integration, Network Infrastructure
- Industrial Networks: Industry 4.0, IIoT, Smart Manufacturing
- Network Management: Monitoring, Configuration, Automation
- Disaster Recovery and Business Continuity Planning: Strategies, Best Practices

Text Books:

1.	Computer Networking-A Top-Down approach, 5thedition, Kurose and Ross, Pearson

Reference Books

1.	Computer Networks (4th edition), Andrew Tanenbaum, Prentice Hall
2.	Computer Networking and the Internet (5thedition),Fred Halsall, Addison Wesley
3.	Data Communications and Networking (4th edition), Behrouz Forouzan, McGraw Hill

Course Code:	SCCSC-502	Compiler Designing	Credits: 4
Course Objectives:			
• To understand the phases of Compiler • To aware the Finite Automata and Lexical Analysis • To understand Parsing Techniques			
Course Outcome:			
• Students will be able to perform Syntax as well as Semantic analysis • Students will be able to plot transition diagrams for DFA and NFA			
Unit-1:	**Introduction**		
	Compiler Basics, Issues in Compilation, Phases of Compilation: the Analysis – Synthesis Model, Compiler Construction Tools		**15 Lectures**
Unit-2:	**Designing a lexical Analyser**		
	Role of Lexical Analysis, Input Buffering, Specification of Tokens, Recognition of Tokens, Regular Expression: Definition, Examples, & Identities, Finite Automata: Concept, DFA: Definition & examples, NFA: Definition, examples, Language accepted by FA, NFA with ε moves, Regular Expression to FA: Method and Problems, NFA with ε moves to NFA, NFA to DFA: Method Problems, Minimization of DFA: Problem using Table, Method - FA with output.		**15 Lectures**
Unit-3:	**Designing a syntax analyzer**		
	Role of Syntax Analyzer, Classification of parsers, Top-Down Parsing: Introduction, Problems in top-down parsing, Recursive Parsing, Problems in Recursive Procedures, Predictive Parsing, Error Handling in Predictive Parsers, Bottom Up Parsing: Shift Reduce Parser, Actions of shift reduce parser, parse tree, Operator Precedence Parsing, Components of operator precedence parsers, operator precedence parsers, Advantages and disadvantages of operator precedence Parsing. LR Parsing: Simple LR parser, LALR parser.		**15 Lectures**
Unit-4:	**Intermediate-Code Generation and Code Optimization**		
	Need For Intermediate Code Generation, syntax trees, three-address code, translation of expressions, stack allocation of space, heap management, storage allocation for arrays, strings and records, Intermediate Forms: Polish Notation, Quadruples, Triples. Introduction and need for code optimization, Optimization with in Basic Blocks: Folding, Redundant operation elimination, Optimization with in Loop: Strength Reduction, Dead code elimination, Moving operation within block out of block.		**15 Lectures**
Prescribed Book			
1.	Aho A.V., R. Sethi and J.D. Ullman, Compiler Principle, Techniques and Tools , Addison Wesley.		
Reference Books			
1.	John E. Hopcroft , Rajeev Motwani, Jeffrey D. Ullman , "Introduction to Automata Theory, Languages and Computation" , Pearson education 2nd Ed		
2.	K.L.P.Mishra & N. Chandrasekaran, " Theory of Computer Science (Automata Languages And Computation)" , PHI 2nd Ed.		
3.	Barret, Couch, Compiler Construction Theory and Practice, , Asian Student Edition		
4.	Dhamdhere D.M, "Compiler Construction Principle and Practice", McMillan India		

Course Code:	SCCSC-503	Software Engineering	Credits: 4
Course Objectives:			
• To aware the software Engineering principles • To understand the software development and testing process, verification and validation of software product			
Course Outcome:			
• Students are able to develop software using software engineering principles • Students are able to test the software as well as quality of software product			
Unit-1:	**Introduction**		
	Introduction: Software, Software Characteristics, Software Applications, Software Myths, Software Engineering, Generic View of Software, Software Paradigms: Linear Sequential Model or Classic Life Cycle, Prototyping, Evolutionary Software Process Model, 4 GT, RAD		15Lectures
Unit-2:	**Software process and project metrics**		
	Measures, metric and indicators, Software Measurement, Reconciling different metric approaches, Metrics for software quality, Integrating metrics within the software Engineering process.		15 Lectures
Unit-3:	**Design Engineering and User Interface Design**		
	Design process and design quality, design concepts, design models, The Golden rules, User interface analysis and design, interface design activities		15 Lectures
Unit-4:	**Risk Analysis AND Software Testing Techniques**		
	Definition of Risk, Types of Risk, Reactive Vs. proactive risk, Risk Analysis: risk identification, projection, Assessment and Management. Testing-Introduction to Testing, Testing Objectives, white box: Basis path Testing, Control Structure Testing, black box: Equivalence Partitioning, Boundary Value Analysis, Comparison Testing, Orthogonal Array, Testing Strategies: Validation and Verification.		15 Lectures
Prescribed Book			
1.	Software Engineering a Practitioner's Approach Roger S. Pressman 5e		
Reference Books			
1.	Software Engineering Richard Fairley Tata McGraw Hill		
2.	Software Engineering David Gustafson		
3.	Practical Guide in Structured System Design Meilier Page		
4.	Software Project Management - Jalote		

Course Code:	SCCSE-501-A	Advanced Python Programming	**Credits:** 03 Theory and 01 Lab	Hours required
Course pre-requisite:				
1. Basic knowledge of Python Programming				
Course Objectives:				
1. Understanding of the Python language and explore advanced concepts and techniques. 2. To master Python and dedicated to pursuing careers as data analysts or data scientists. **3.** Providing students with a strong foundation in programming and data analysis, focusing on data analysis, visualization, and machine learning.				
Course Outcome:				
1. Students will be able to use different python modules. 2. Students will be able to perform basic data analytics operations. 3. Student wiil be able to implement Machine Learning algorithms.				
Unit -1	**Advanced Data Structures and Algorithms**			
1.1	Review of basic data structures: lists,dictionaries,sets,tuples , Advanced data structures: trees,graphs, heaps, queues			6
1.2	Algorithms for searching, sorting, and graph traversal, Big O notation and algorithmic complexity analysis			**6**
Unit-2	**Data analysis using Python**			
2.1	Introduction to Data Analysis,NumPy:Arrays and Array Operations, Array Indexing,Slicing, Broadcasting,Vectorization,ArrayManipulation,Reshaping			**6**
2.2	Introduction to Pandas, Pandas data structures, Data wrangling using pandas,Loading a dataset into a dataframe,			**6**
2.3	Selecting Columns from a dataframe,Selecting Rows from a dataframe, Adding new data in a dataframe,Deleting data from a dataframe, Handling Missing Data			**6**
Unit-3	**Data Exporation and visualization unsing Python**			
3.1	Exporatory Data Analysis (EDA),Introduction to Matplotlib,Scatter plot, Line plot, Bar chart, Histogram,Box plot			**10**
3.2	Web Scrapping, Beautiful Soup			**4**
Unit-4	**Machine learning with scikit-learn**			
4.1	Basic of Machine Learning,Linear Regression,Logistic Regression			**6**
4.2	Unsupervised Learning,Model Evaluation parameters			**6**
4.3	Hyperparameter Tuning and Model Selection			**4**
Text Book: 1. Fluent Python" by Luciano Ramalho 2. Python for Data Analysis by Wes McKinney 3. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurelien Geron				
Reference Books: 1. Data Structures and Algorithms in Python by Michael T. , Roberto T, and Michael H. G. 2. Python data science handbook by Jake Vanderplas				

Course Code:	SCCSE-501-B	Network and Linux Administration	**Credits:** 03 Theory and 01 Lab
Course Objectives:			
1. To understand File formats and data standard used in Multimedia 2. To acquire the skill of animation of an object 3. To study the Input output technology of Multimedia			
Course Outcome:			
1. Students will be able to Concept of File formats 2. Students will be able to do the animation 3. Students will able to compress the data used in multimedia without losing the properties			
Unit-1:	**Introduction**		
	Basics of Client and Server model, basic networking topologies , Installation of Linux, System recovery, File system, system calls, internal commands of Linux: Date, Time,cp, cal, rd, md, cd		15 Lectures
Unit-2:	**Component of Process AND File System**		
	PID, PPID, UID, EUID, GID, EGID, The lifecycle of Process, The /Proc file system, The working of commands top, nice ,renice, ps, dig File system mounting and unmounting, File types: regular files, directories, character and block device files, names pipes. File attributes: permission bits, setuid and set gid bits,		15 Lectures
Unit-3:	**Linux administration**		
	Adding user, removing user, disable login, allocating permissions to user, managing user with system specific tools. Software Configuration Management: diskless client, Package management, Localization and configuration, configuration management tools. Linux commands: grep, man, kill, whereis, service,df,du,passwd,lpr,ifconfig,netstat,nslookup,wall, talk,free, cat, tar,		15 Lectures
Unit-4:	**Domain Name System (DNS) in Linux**		
	DNS namespace, How DNS works, DNS database: Resource record, SOA record, NS record, Mx record, PTR record, Cname record, IPV6 resource record. BIND client issues, BIND server configuration,		15 Lectures needed
Prescribed Book			
1.	Evi Nemeth , Garth Snyder, Trent R. Hein, Ben Whaley "Unix and Linux administration handbook" 4th Ed. ,PHI		
Reference Books			
1.	Evi Nemeth , Garth Snyder, Trent R. Hein "Unix and Linux administration handbook" 2th Ed. ,PHI		

Course Code:	SCCSE-501-C	DBMS PROGRAMMING	**Credits:** 03 Theory and 01 Lab
Course Objectives:			
1. Make students familiar with triggers, procedures, stored procedures in SQL			
Course Outcome:			
1. Students will able to understand the use of programming in DBMS 2. Students will able to develop their own procedures in SQL			
Unit-1:	**Introduction AND Basics of PL/SQL**		
	Overview of PL/SQL and its features , Differences between SQL and PL/SQL, PL/SQL architecture and components (blocks, procedures, functions, packages), PL/SQL block structure (declarative, executable, exception handling), Variables and data types, Operators and expressions		15 Lectures
Unit-2:	**Control Structures and Cursors in PL/SQL**		
	Iterative control: LOOP, WHILE, FOR loops, Implicit vs. Explicit cursors, Cursor attributes (%ROWCOUNT, %FOUND, % NOTFOUND, %ISOPEN), Cursor FOR loops and cursor parameters		15 Lectures
Unit-3:	**PL/SQL Subprograms and Exception Handling**		
	Exception types (predefined and user-defined), Handling exceptions with EXCEPTION and RAISE statements , Exception propagation and nested blocks, - Procedures and functions: creation, parameters (IN, OUT, IN OUT), Invoking subprograms from SQL and PL/SQL, Overloading subprograms		15 Lectures
Unit-4:	**Packages and Triggers in PL/SQL**		
	Introduction to packages: specification and body, Advantages of using packages, Using package variables and procedures /functions, Introduction to triggers, Types of triggers (BEFORE, AFTER, INSTEAD OF), Trigger timing and event-based triggers		15 Lectures
Prescribed Book			
1.	Oracle PL/SQL Programming" by Steven Feuerstein		
Reference Books			
1.	Oracle Database PL/SQL Language Reference		
2.	Online resources: Oracle documentation, PL/SQL tutorials, SQL*Plus guides		

Course Code:	**SCCSE-501-D**	**Theory of Computations**	**Credits:** 03 Theory and 01 Lab
Course Objectives:			
• The learning objectives of this course are to introduce students to the mathematical foundations of computation including automata theory, the theory of formal languages and grammars, the notions of algorithm, decidability, complexity, and computability, enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms			
Course Outcome:			
1. Ability to prove results using proof by induction, proof by contradiction, proof by construction, proof by case exhaustion 2. Understanding of regular and context-free languages. 3. Ability to describe and transform regular expressions and grammars 4. Understanding of the key results in algorithmic complexity, computability and solvability of problems.			
Unit-1:	**Finite Automata and Regular Expressions:**		
	Definition of Deterministic Finite Automata, Non Deterministic Finite Automata, Moore and Mealy Machines and their conversions, Regular Expressions, recursive definition, NFA with e -Moves, Inter conversion between NFA and DFA and DFA regular expression and FA, Pumping lemma.		15 Lectures
Unit-2:	**Context Free Grammars**		
	Definition, production rules, ambiguous grammar, removal of ambiguity, Chomsky hierarchy, Context Free Grammar (CFG) - definition simplification of CFG.		15 Lectures
Unit-3:	**Context Free Languages**		
	Definition of Context free Languages, regular grammar definition, left linear right linear grammar, Inter conversion between left linear and right linear regular grammar, Regular grammar and finite automata, CNF, GNF, derivation graphs type0 and type1 grammars		15 Lectures
Unit-4:	**Pushdown automata and Turing Machines**		
	Formal definition, Pushdown automata (PDA), deterministic pushdown automata (DPDA) –definition, non- deterministic pushdown automata (NPDA)-definition relative powers of DPDA and NPDA. The definition of a Turing machine, computing with Turing machine, Extensions of Turing machines, Random access Turing machines, Non-deterministic Turing machines		15 Lectures
Prescribed Book			
1.	Hopcroft, Ullman, Introduction to Automata Theory, Languages, and Computation, Addison Wesley Pub		
Reference Books			
1.	Daniel I. A. Cohen, Introduction to computer theory, Willey Pub.		
2.	John C. Martin, Introduction to Languages and Theory of Computation, McGraw Hill		

Course Code:	**SCCSE-501-E**	**Internet of Things**	**Credits:** 03 Theory and 01 Lab
Course Objectives:			
• Understanding IoT details and technology needed to build IoT applications			
Course Outcome:			
• Students will be able to design their own IoT solutions			
Unit-1:	**Introduction to IoT**		
	1. Overview of IoT: Definition, History, and Evolution , IoT Architecture: Layers and Design Principles, IoT Ecosystem: Devices, Gateways, and Cloud 2. Key Technologies: Sensors, Actuators, and Communication Protocols, IoT Network Protocols: IPv6, 6LoWPAN, RPL, CoAP, MQTT 3. Connectivity Technologies: WiFi, Bluetooth, Zigbee, LoRaWAN, NB-IoT, IoT Standards and Frameworks: IEEE, IETF, ITU-T, OneM2M, Case Studies: Smart Homes, Wearables, Industrial IoT		15 Lectures
Unit-2:	**IoT Software and Hardware**		
	1. Introduction to IoT Hardware Platforms: Arduino, Raspberry Pi, ESP8266/ESP32, Sensors and Actuators: Types, Selection Criteria, and Interfacing, IoT Software Platforms and Operating Systems: FreeRTOS, Contiki, RIOT 2. IoT Programming Languages: Python, C/C++, JavaScript, Data Collection and Processing: Edge vs Cloud Computing, IoT Data Storage: Time Series Databases, NoSQL, SQL 3. Introduction to IoT Analytics: Descriptive, Predictive, and Prescriptive Analytics, Visualization Tools and Techniques: Dashboards, Reports, and Alerts		15 Lectures
Unit-3:	**IoT communication and Security**		
	1. IoT Communication Models: Device-to-Device, Device-to-Gateway, Device-to-Cloud, Network Topologies: Star, Mesh, Hybrid 2. IoT Security Fundamentals: Threats, Vulnerabilities, and Attack Surfaces, Cryptographic Techniques for IoT Security: Symmetric and Asymmetric Cryptography, PKI, Secure Communication Protocols: TLS/SSL, DTLS, IPsec, Authentication and Authorization in IoT: OAuth, JWT, X.509 Certificates 3. Privacy Considerations in IoT: Data Privacy Laws, GDPR, CCPA 4. Security Best Practices: Device Hardening, Network Security, Incident Response		15 Lectures
Unit-4:	**IoT application and Future Trends**		
	1. IoT in Smart Cities: Smart Grid, Smart Transportation, Smart Healthcare, Industrial IoT (IIoT): Industry 4.0, Predictive Maintenance, Asset Tracking, IoT in Agriculture: Precision Farming, Smart Irrigation, Livestock Monitoring 2. Emerging Technologies in IoT: Edge Computing, Fog Computing, Blockchain, Artificial Intelligence and Machine Learning in IoT: Use Cases and Techniques 3. Future Trends in IoT: 5G and Beyond, Autonomous Systems, Ethical and Social Implications, Course Review and Capstone Project Presentation		15 Lectures
Prescribed Book			
1.	Internet of Things: A Hands-On Approach by Arshdeep Bahga and Vijay Madisetti		
Reference Books			
1.	Building the Internet of Things by Maciej Kranz		

Course Code:	**SCCSE-501-F**	**Subject relevant MOOC**	**Credits:** 03 Theory and 01 Lab
Subject relevant MOOC (NPTEL / SWAYAM / RUSA sponsored Future Oriented Courses / other recognized courses[$$] ($$ - with prior consultation with the Director of the School in order to determine relevancy, credit transfer policy will be adopted)			

Course Code:	**SCCSE-501-G**	**Visual Programming Tools**	**Credits:** 03 Theory and 01 Lab
Course Objectives:			
This course introduces students to visual programming using Visual Basic, focusing on creating graphical user interfaces (GUIs) for Windows applications. Students will learn to design, code, and test user-friendly applications using various Visual Basic controls and techniques.			
Course Outcome:			
• Students will be able to develop independent applications			
Unit-1:	**Fundamentals of Visual Basic and Visual Studio**		
	Introduction to Visual Programming, Importance of Visual Programming, Visual Basic and Visual Programming, Setting Up the Development Environment, Basic Syntax and Structure of Visual Basic Programs, Variables, Constants, and Data Types, Operators and Expressions, Creating basic GUI Application		15 Lectures
Unit-2:	**Working with Controls and Events**		
	Concept of Form, Common Controls (Labels, TextBoxes, Buttons, CheckBoxes, RadioButtons, etc), Setting Properties of Controls (Text, Font, Color, Size, etc.), Aligning and Arranging Controls on a Form, Event Handling (Understanding Events and Event Handlers, Writing Event Handlers for Controls, Common Event Handlers), User Input Validation, Enhancing User Interfaces - Best Practices for UI Design		15 Lectures
Unit-3:	**Advanced Controls and Customization**		
	Working with ListBoxes, ComboBoxes, and PictureBoxes, Standard DialogBoxes, Menus and Toolbars, Creating Menus with the MenuStrip Control, Adding Toolbars with the ToolStrip Control, Multi-Form Applications		15 Lectures
Unit-4:	**File Handling in Visual Basic**		
	Introduction to File Handling, File Input and Output Operations, Working with Binary Files, Advanced File Handling Techniques :- OpenFileDialog and SaveFileDialog, Working with File Paths and Directories, Using the My.Computer.FileSystem Object		15 Lectures
Prescribed Book			
1.	The Complete Reference Visual basic .NET, Jeffrey R. Shaprio		
Reference Books			
1.	Mastering Visual Basic 2010, Evangelos Petroutsos, Sybex		
2.	3. Programming Windows, Charles Petzold, Microsoft Press		

Course Code:	**SCCSCP-501**	**LAB 7: CN LAB**	**Credits: 01**
Course Objectives: As per the Lab Manual circulated to students by the concerned Teacher			
Course Outcome: As per the Lab Manual circulated to students by the concerned Teacher			
Experiments: As per the Lab Manual circulated to students by the concerned Teacher			

Course Code:	**SCCSCP-502**	**LAB 8: CD LAB**	**Credits: 01**
Course Objectives: As per the Lab Manual circulated to students by the concerned Teacher			
Course Outcome: As per the Lab Manual circulated to students by the concerned Teacher			
Experiments: As per the Lab Manual circulated to students by the concerned Teacher			

Course Code:	SCCSEP-501	LAB 9: ELECTIVE LAB	Credits: 01
Course Objectives: As per the Lab Manual circulated to students by the concerned Teacher			
Course Outcome: As per the Lab Manual circulated to students by the concerned Teacher			
Experiments: As per the Lab Manual circulated to students by the concerned Teacher			

Course Code: SCCSR-551: RESEARCH PROJECT

Course Objectives: To provide postgraduate level research knowledge in computer science, including understanding, analysis, management, and handling of real-life information technology problems in workplace. Students are encouraged to problems from real life / NGO/ state-central govt projects/ hackathon/ etc

Course Outcome: Project based learning will increase their capacity and learning through shared cognition. Students will have an ability to identify, formulate and implement computing solutions. Students will be able to design a system, component or process as per needs and specification.

Guidelines for Project Development:

1. A single student or a group of maximum three students should be formed at the beginning of the semester. University policy will be adopted for number of students in a project group.
2. Each project will be allotted one project guide.
3. Students must submit the project topic and synopsis to the project guide.
4. Students will be given a project approval letter signed by the head of department and the project guide.
5. After receiving a project approval letter, students must submit at least three progress reports of their development in project to the guide, one per month.
6. After completion of project students have to give pre-exam demo to his guide.
7. After finalization of the project, students must prepare minimum 03 copies of the project reports, out of which one copy is for the college and one copy is for the university records. University/College copy must be bind with black covering with golden embossment and it should contain
   i. First Page
   ii. Certificate
   iii. Declaration
   iv. Acknowledgement
   v. Project Approval letter
   vi. Pre- Mid and Final Progress reports (03)
   vii. Objectives, Hypothesis, Methodology and System Flow Diagram/DFD
   viii. Chapter wise briefing, results, conclusions, snapshots, code, etc
   ix. Bibliography
   x. Selected publications as research papers or book chapters or case studies / etc