Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Compiler Design**	Course Code: **C109611(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 3 T: 1 P: 0** Credits: **4**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course Objectives:

- Course objective is to make the student to understand the process involved in a compiler, create an overall view of various types of translators, linkers, loaders, and phases of a compiler, understand what is syntax analysis, various types of parsers especially the top down approach, awareness among students the various types of bottom up parsers, understand the semantic analysis and intermediate code generation, the role of symbol table and its organization, Code generation, machine dependent code optimization.

UNIT-I: Introduction

Introduction to Compiler, single and multi-pass compilers, Translators, Phases of Compilers, Compiler writing tools, Finite Automata and Lexical Analyzer: Role of Lexical Analyzer, Specification of tokens, Recognition of tokens, Regular expression, Finite automata, from regularexpression to finite automata, transition diagrams, Implementation of lexical analyzer with LEX.

UNIT-II: Syntax Analysis and Parsing Techniques

Context free grammars, Bottom-up parsing and top down parsing, Top down Parsing : elimination of left recursion, recursive descent parsing, Predicative Parsing, Bottom Up Parsing : Operator precedence parsing, LR parsers, Construction of SLR, Canonical LR and LALR parsing tables, Construction of SLR parse tables for Ambiguous grammar, parser generator- YACC, error recovery in top down and bottom up parsing.

UNIT-III: Syntax Directed Translation & Intermediate Code Generation

Synthesized and inherited attributes, Construction of syntax trees, bottom up and top down evaluation of attributes, S- attributed and L-attributed definitions ,Postfix notation; Three address codes, quadruples, triples and indirect triples, Translation of assignment statements, control flow,Boolean expression and Procedure Calls.

UNIT-IV: Run-time Environment

Storage organization, activation trees, activation records, allocation strategies, Parameter passing symbol table, dynamic storage allocation.

UNIT-V:

Code Optimization and Code Generation

Basic blocks and flow graphs, Optimization of basic blocks, Loop optimization, Global data flow analysis, Loop invariant computations. Issue in the design of Code generator, register allocation, the target machine, and simple Code generator.

Text Books:

1. Compilers Principles, Techniques and Tools, Alfred V. Aho, Ravi Sethi and Ullman J.D., Addison Wesley.
2. Principle of Compiler Design, Alfred V. Aho and J.D. Ullman, Narosa Publication.
3. Introduction to CompilerTechniques, J.P. Bennet, Tata McGraw-Hill.

Reference Books:

1. Compiler Design in C, A.C. Holub, PHI.
2. Compiler construction (Theory and Practice), A. Barret William andR.M., Bates, Galgotia Publication.
3. Compiler Design, O.G. Kakde, Laxmi Publication.

Course Outcomes: On successful completion of the course, the student will be able to:

- Explain the concepts of Compilers and roles of the lexical analyzer.
- Apply the concepts of different Parsing techniques and implement the knowledge to Yacc tool.
- Develop syntax directed translation schemes.
- Implement the principles of scoping, parameter passing and runtime memory management.
- Use the new code optimization techniques to improve the performance of a program in terms of speed & space and develop algorithms to generate code for a target machine.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Software Engineering and Project Management**	Course Code: **C109613(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 3 T: 1 P: 0** Credits: **4**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course objectives:

- To provide knowledge to identify problems associated with conventional software development process. models and solving through modern component based software development process models.
- To provide knowledge to develop software requirement specification.
- To represent workflow environment in the software industry.
- To provide knowledge to estimate various checkpoints during software development.
- To provide comprehensive knowledge for software project development and assessment.

UNIT-I: Introduction to software engineering- scope of software engineering, — historical aspects, economic aspects, maintenance aspects, specification and design aspects, team programming aspects. Software engineering a layered technology — processes, methods and tools. Software process models – prototyping models, incremental models, spiral model, waterfall model. ProcessFramework Models: Capability maturity model (CMM), ISO 9000. Phases in Software development – requirement analysis- requirements elicitation for software, analysis principles, software prototyping, specification.

UNIT-II: Types of Requirements, Feasibility Study, Requirement Analysis and Design: DFD, Data Dictionary, HIPO Chart, Warnier Orr Diagram, Requirement Elicitation: Interviews, Questionnaire, Brainstorming, Facilitated Application Specification Technique (FAST), Use Case Approach. SRS Case study, Software Estimation: Size Estimation: Function Point. Cost Estimation: COCOMO.

UNIT-III: Planning phase — project planning objective, software scope, empirical estimation models-COCOMO, single variable model, staffing and personal planning. Design phase – design process, principles, concepts, effective modular design, top down, bottom-up strategies, stepwise refinement. Coding – programming practice, verification, size measures, complexity analysis, coding standards. Testing — fundamentals, white box testing, control structure testing, black box testing, basis path testing, code walk-throughs and inspection, testing strategies-Issues, Unit testing, integration testing, Validation testing, System testing. Maintenance-Overview of maintenance process, types ofmaintenance. Risk management: software risks - risk identification-risk monitoring and management.

UNIT-IV: Project Management concept: People – Product-Process-Project. An Overview of IT Project Management: Define project, project management framework, The role of project Manager, Systems View of Project Management, Stakeholder management. Project scheduling and tracking. Project Scope management: Scope definition and Project Scope management, Creating the Work Breakdown Structures, Scope Verification, Scope Control.

UNIT-V: Software configuration management: Basics and standards User interface design - rules. Computeraided software engineering tools - CASE building blocks, taxonomy of CASE tools, integrated CASE environment.

Text Books:

1. Software Engineering, Roger S Pressman, McGraw Hill publication.
2. Software Engineering, Ian Sommerville, Pearson Education.
3. Software Project Management: Aunified frame work, Walker Royce, Pearson Education.

Reference Books:

1. Software Engineering, K.K. Aggarwal and Yogesh Singh, New age International Publishers.
2. Software Project Management: Aconcise study, S.A. Kelkar, PHI.
3. Software Engineering Project Management, Richard H. Thayer, Wiley India Publication.

Course outcomes:

- The student will have a fair idea about the importance of using software engineering principles in real life projects.
- The student will be able to prepare software requirement sheet for a real life project, keeping in mind the properties of an SRS document.
- The student will be able to use mathematical models for calculating the size, cost and duration of real life projects.
- Apply the concepts of project planning and management.
- The student will be able to develop projects using standard procedures and CASE tools.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Cryptography and Network Security**	Course Code: **C131612(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 3 T: 1 P: 0** Credits: **4**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course Objectives:

- To understand the principles and practices of cryptography and network security.
- To understand the practical applications that have been implemented and are in use to provide network security.

UNIT-I: Introduction: Security trends, Legal, Ethical and Professional Aspects of Security, Need for Security at Multiple levels, Security Policies, Model of network security, Security attacks, services and mechanisms, OSI security architecture, Classical encryption techniques: substitution techniques, transposition techniques.

UNIT-II: Symmetric Ciphers: MATHEMATICS OF SYMMETRIC KEY CRYPTOGRAPHY: Algebraic structures, Modular arithmetic, Euclid's algorithm, Congruence and matrices, Groups, Rings, Fields, Finite fields, SYMMETRIC KEY CIPHERS: DES, Block cipher Principles of DES, Strength of DES, Differential and linear cryptanalysis, Block cipher design principles, Block cipher mode of operation, Evaluation criteria for AES, Advanced Encryption Standard, RC4, Key distribution.

UNIT-III: Asymmetric (Public Key) Ciphers: MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes, Primality Testing, Factorization, Euler's totient function, Fermat's and Euler's Theorem, Chinese Remainder Theorem, Exponentiation and logarithm, ASYMMETRIC KEY CIPHERS: RSA cryptosystem , Key distribution, Key management, Diffie Hellman key exchange, Elliptic curve arithmetic, Elliptic curve cryptography.

UNIT-IV: Asymmetric Ciphers (continued): Message Authentication and Hash functions: Authentication requirement, Authentication function, MAC, Hash function, Security of hash function and MAC, SHA, Digital signature and authentication protocols, DSS, Authentication applications, Kerberos, X.509

UNIT-V: Network Security applications: Authentication applications: Electronic mail security: PGP, S/MIME. Overview of IP Security. Web Security: Web security considerations, SSL and TLS, Secure electronic transaction. System Security: Intruders, Intrusion detection, password management, viruses and related threats, virus counter measures.

Text Books:

1. Cryptography and Network Security: Principles and Practice" by William Stallings, Pearson Education, Prentice Hall, 4th Edition.
2. Cryptography and Network Security, Atul Kahate, McGraw Hill Education (India) Private Limited; Third edition.

References:

1. Applied Cryptography: Protocols & Algorithms, Schneier & Bruce, MGH International.
2. Cryptography and Security, by Dr T R Padmanabhan N Harini , Wiley India Pvt Ltd, 2011.
3. Handbook of Applied Cryptography" by Alfred J Menezes and Scott A Vanstone.
4. Cryptography and Network Security" by Behrouz A Forouzan and Debdeep Mukhopadhyay.
5. Network Security: The Complete Reference, Tata McGraw-Hill Education, by Roberta Bragg, Mark RhodesOusley, Keith Strassberg.

Course Outcome: After successful completion of this course, the students will be able to explain

1. Conventional encryption algorithms for confidentiality and their design principles.
2. Public key encryption algorithms and their design principles.
3. Use of message authentication codes, hash functions, digital signature and public key certificates.
4. Network security tools and applications.
5. System-level security issues like threat of and countermeasures for intruders and viruses, and the use of firewalls and trusted systems.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Machine Learning *(Professional Elective - II)***	Course Code: **C131631(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 2 T: 1 P: 0** Credits: **3**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course Objectives:

- To introduce the basic concepts of Machine Learning.
- To introduce the concepts of Data Analysis in Machine Learning.
- To introduce the concepts of Machine Learning algorithms.
- To introduce the concepts related to Machine Learning model diagnosis and tuning.
- To introduce the concepts related to Artificial Neural Networks.

UNIT- I: Introduction: History and Evolution, Machine Learning Categories: Supervised Learning, Unsupervised Learning, Reinforcement Learning. Knowledge Discovery in Databases, SEMMA (Sample, Explore, Modify, Model, Assess).

UNIT-II: Machine Learning Perspective of Data: Scales of Measurement, Dealing with Missing Data, Handling Categorical Data, Normalizing Data, Feature Construction or Generation. Correlation and Causation, Polynomial Regression, Logistic Regression, ROC Curve.

UNIT- III: Introduction to Machine Learning Algorithms: Decision Trees, Support Vector Machine, k-Nearest Neighbors, Time-Series Forecasting, Clustering, Principal Component Analysis (PCA).
UNIT-IV Model Diagnosis and Tuning: Bias and Variance, K-Fold Cross Validation, Bagging, RandomForest, Gradient Boosting, Stacking.

UNIT-V: Artificial Neural Network (ANN): Perceptron—Single Artificial Neuron, Multilayer Perceptrons (Feedforward Neural Network), Restricted Boltzman Machines (RBMs).

Text Books:

1. Mastering Machine Learning with Python in Six Steps– Manohar Swamynathan, APress.
2. Python Machine Learning for Beginners, M. Usman Malik, AI Publishing.

Reference Books:

1. Introduction to Machine Learning with Python, Daniel Nedal& Peters Morgan, AI Sciences
2. Data Structures and Algorithms Using Python, Rance D. Necaise, WILEY

Course Outcomes [After undergoing the course, students will be able to:]

- Remember the basic concepts of Machine Learning.
- Able to perform Data Analysis in Machine Learning.
- Remember the Machine Learning algorithms.
- Able to perform Machine Learning model diagnosis and tuning.
- Apply the concept of Artificial Neural Networks.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Network Management Systems and Operations** ***(Professional Elective - II)***	Course Code: **C131632(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 2 T: 1 P: 0** Credits: **3**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course Objectives:

- To maintain optimal network performance and availability, and to ensure continuous uptime.
- Monitor the network for problems that require special attention.

UNIT- I:

The Network Management Challenge: Introduction, The Internet and Network Management, Internet Structure, Managing an Entity, Internal and External policies, The state of Network Management, Network Management in the Gartner Model, Benefits of Automation, The Lack of Industry Response, Impact on Business, Distributed Systems and new abstractions. A Review of Network Elements and Services: Introduction, Network Devices and Network Services, Network Elements and Element Management, Effect of physical organization on Management, Examples of Network Elements and Services, Basic Ethernet Switch, VLAN Switch, Access Point for a Wireless LAN, Cable Modem System, DSL Modem System and DSLAM, CSU/DSU used in Wide Area Digital Circuits, Channel Bank, IP Router, Firewall, DNS Server, DHCP Server, Web Server, HTTP Load Balancer.

UNIT- II:

The Network Management Problem: Introduction, What is Network Management?, The scope of Network Management, variety and multi-vendor environments, element and network management systems, scale and complexity, types of networks, classification of devices, FCAPS: The Industry Standard Definition, The motivation for automation, Why Automation has not occurred, Organization of management Software. Configuration and Operation: Introduction, Intuition for configuration, configuration and protocol layering, dependencies among configuration parameters, seeking a more precise definition of configuration, configuration and temporal consequences, configuration and global consistency, global state and practical systems, configuration and default values, partial state, automatic update and recovery, Interface paradigm and incremental configuration, commit and rollback during configuration, automated rollback and timeout, snapshot, configuration, and partial state, separation of setup and activation.

UNIT- III:

Fault detection and correction: Introduction, Network Faults, Trouble Reports, Symptoms, And Causes, Troubleshooting And Diagnostics, Monitoring, Baselines, Items That Can Be Monitored, Alarms, Logs, And Polling, Identifying The Cause Of A Fault, Human Failure And Network Faults, Protocol Layering And Faults, Hidden Faults And Automatic Correction, Anomaly Detection And Event Correlation, Fault Prevention. Performance Assessment and Optimization: Introduction, aspects of performance, Items that can be measured, measures of network performance, application and endpoint sensitivity, degraded service, variance in traffic and congestion, congestion, delay and utilization, local and end-to-end measurements, passive observation Vs. active probing, bottlenecks and future planning, capacity Planning, planning the capacity of a switch, planning the capacity of a router, planning the capacity of an Internet connection, measuring peak and average traffic on a link, estimated peak utilization and 95th percentile, relationship between average and peak utilization, consequences for management and the 50/80 Rule, capacity planning for a complex topology, a capacity planning process, route changes and traffic engineering, failure scenarios and availability.

UNIT- IV:

Security: Introduction, The illusion of a secure network, security as a process, security terminology and concepts, management goals related to security, Risk Assessment, Security policies, acceptable use policy, basic technologies used for security, management issues and security, Security architecture: Perimeter Vs. Resources, element coordination and firewall unification, resource limits and denial of service, management of authentication, access control and user authentication, management of wireless networks, security of the network, role-based access control, audit trails and security logging, key management.

UNIT-V:

Management tools and technologies: Introduction, the principle of most recent change, the evolution of Management tools, management tools as applications, using a separate network for management, types of management tools, physical layer testing tools, reach ability and connectivity tools (ping), packet analysis tools, discovery tools, device interrogation interfaces and tools, event monitoring tools, triggers, Urgency Levels, And Granularity, events, Urgency Levels and traffic, performance monitoring tools, flow analysis tools, routing and traffic engineering tools, Configuration tools, Security Enforcement tools, Network Planning tools, Integration of Management tools, NOCs and Remote Monitoring, Remote CLI Access, Remote Aggregation Of Management Traffic. Network Management Tools: Zabbix Labs, Nagios, Google Cloud network, Automation with Terraform.

Text Books:

1. Automated Network Management Systems, D. Comer, Prentice Hall, 2006, ISBN No. 0132393085.
2. Nagios Core Administration Cookbook - Second Edition, Tom Ryder, 2016, Packt publishing, ISBN: 781785889332.
3. Terraform: Up and Running, Yevgeniy Brikman, 2017, O'Reilly Media, Inc., ISBN: 9781491977088.

Reference Book:

1. Applied Network Security Monitoring, Chris Sanders, Jason Smith, Syngress publications.

Course Outcomes:

- Understand the basic network elements and their services.
- To able to familiarize with different network faults and their correction techniques.
- Understand various measures of network performance.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Linux Programming *(Professional Elective - II)***	Course Code: **C131633(022)**
Total / Minimum-Pass Marks (End Semester Exam): 100 / 35	**L: 2 T: 1 P: 0** Credits: **3**
Class Tests & Assignments to be conducted: **2 each**	**Duration (End Semester Exam): 03 Hours**

Course Objectives:

- To teach principles of operating system including File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking Commands, Basic Linux commands, Scripts and filters.
- To familiarize fundamentals of the Bourne again shell (bash), shell programming, pipes, input and output redirection Control structures, arithmetic in shell interrupt processing, functions, debugging shell scripts.
- To impart fundamentals of file concepts kernel support for file, File structure related system calls (file API's).
- To facilitate students in understanding Inter process communication.
- To facilitate students in understanding semaphore and shared memory.
- To facilitate students in understanding process.

UNIT–I: INTRODUCTION TO LINUX AND LINUX UTILITIES: A brief history of LINUX, architecture of LINUX, features of LINUX, introduction to vi editor. Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin.Text Processing utilities and backup utilities , tail, head , sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio.

UNIT–II: Introduction to Shells: Linux Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command-Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization. Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files.

UNIT–III: Grep: Operation, grep Family, Searching for File Content. Sed :Scripts, Operation, Addresses, commands, Applications, grep and sed. UNIX FILE STRUCTURE: Introduction to UNIX file system, inode (Index Node), file descriptors, system calls and device drivers. File Management :File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir, umask.

UNIT–IV: PROCESS AND SIGNALS: Process, process identifiers, process structure: process table, viewing processes, system processes, process scheduling, starting new processes: waiting for a process, zombie processes, orphan process, fork, vfork, exit, wait, waitpid, exec, signals functions, unreliable signals, interrupted system calls, kill, raise, alarm, pause, abort, system, sleep functions, signal sets. File locking: creating lock files, locking regions, use of read and write with locking, competing locks, other lock commands, deadlocks.

UNIT–V: INTER PROCESS COMMUNICATION: Pipe, process pipes, the pipe call, parent and child processes, and named pipes: fifos, semaphores: semget, semop, semctl, message queues: msgget, msgsnd, msgrcv, msgctl, shared memory: shmget, shmat, shmdt, shmctl, ipc status commands. INTRODUCTION TO SOCKETS: Socket, socket connections - socket attributes, socket addresses, socket, connect, bind, listen, accept, socket communications.

Text Books:

1. W. Richard. Stevens (2005), Advanced Programming in the UNIX Environment, 3rd edition, Pearson Education, New Delhi, India.
2. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg.Thomson

References:

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Advanced Programming in the UNIX environment, 2nd Edition, W.R.Stevens, Pearson Education.
3. UNIX Network Programming, W.R. Stevens, PHI. UNIX for Programmers and Users, 3rd Edition, Graham Glass, King Ables, Pearson Education

Course Outcomes:

- Ability to use various Linux commands that are used to manipulate system operations at admin level and a prerequisite to pursue job as a Network administrator.
- Ability to write Shell Programming using Linux commands.
- Ability to design and write application to manipulate internal kernel level Linux File System.
- Ability to develop IPC-API's that can be used to control various processes for synchronization.
- Ability to develop Network Programming that allows applications to make efficient use of resources available on different machines in a network.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: B.Tech (VI Sem)	Branch: Cyber Security
Subject: Cryptography and Network Security (Lab)	Course Code: C131621(022)
Total / Minimum-Pass Marks (End Semester Exam): 40 / 20	L: 0 T: 0 P: 2 Credits: 1

Laboratory Objectives:

- To learn different cipher techniques.
- To implement the algorithms DES, RSA, MD5, SHA-1.
- To use network security tools and vulnerability assessment tools.

List of Experiments:

1. Perform encryption, decryption using the following substitution techniques

   (i) Ceaser cipher, (ii) Playfair cipher iii) Hill Cipher iv) Vigenere cipher
2. Perform encryption and decryption using following transposition techniques

   i) Rail fence ii) row & Column Transformation
3. Apply DES algorithm for practical applications.
4. Apply AES algorithm for practical applications.
5. Implement RSA Algorithm using HTML and JavaScript
6. Implement the Diffie-Hellman Key Exchange algorithm for a given problem.
7. Calculate the message digest of a text using the SHA-1 algorithm.
8. Implement the SIGNATURE SCHEME – Digital Signature Standard.
9. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w.
10. Automated Attack and Penetration Tools Exploring N-Stalker, a Vulnerability Assessment Tool
11. Defeating Malware

    i) Building Trojans ii) Rootkit Hunter

Laboratory Outcomes:

Upon Completion of the course, the students will be able to:

- Develop code for classical Encryption Techniques to solve the problems.
- Build cryptosystems by applying symmetric and public key encryption algorithms.
- Construct code for authentication algorithms.
- Develop a signature scheme using Digital signature standard.
- Demonstrate the network security system using open source tools.

References:

1. Build Your Own Security Lab, Michael Gregg, Wiley India

List of Equipment for a Batch of 30 Students: Software:

C / C++ / Java or equivalent compiler GnuPG, Snort, N-Stalker or Equivalent.

Hardware:

Standalone desktops – 30 Nos. (or) Server supporting 30 terminals or more.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Android (Lab)**	Course Code: **C131622(022)**
Total / Minimum-Pass Marks (End Semester Exam): 40/20	**L: 0 T: 0 P: 2** Credits: **1**

Laboratory Objectives:

- Understand how Android applications work, their life cycle, manifest, Intents, and using external resources
- Design and develop useful Android applications with compelling user interfaces by using, extending, and creating your own layouts and Views and using Menus.
- Secure, tune, package, and deploy Android applications
- Use Android's communication APIs for SMS, telephony, network management, and internet resources (HTTP).

List of Experiments:
(Each student should perform, at least, 10 experiments.)

1.
a) Create an android application to display CSVTU Text Message.
b) Create an android application to display CSVTU Message by using Button.

2. Create an android application to call different activities by using Implicit and Explicit Intents.

3.
a) Create an android application to select item from given list by using AutoCompleteTextView (ACTV).
b) Create an android application to display dropdown menu items and pick one item by using Spinner Component.

4.
a) Create an android application to display internal storage data using Array Adapter.
b) Create an android application to display internal storage data in vertical format by using Custom Adapter.

5. Create an android application to display WhatsApp videos in grid view by using Custom Adapter.

6. Create an android application to display webpage by using Web view Component.

7. Create an android application to display different webpages in fragments by using Fragments Component.

8. Create an android application to store the data by using Shared Preferences.

9. Create an android application to demonstrate concept of SQLite Database Storage method

10. Create an android application to perform different types of operations (Send SMS, Making call and sending email) by using Telephony app.

11. Write an android program to develop Media player application.

12.
a) Write an android program to develop Video view application
b) Write an android program to develop Audio Recording application.

13.
a) Write an android program to develop Video Recording application.
b) Write an android program to develop Camera and Gallery application.

14.
a) Create an android application to get latitude and longitude value by using Location Service.
b) Create an android application to display X, Y Sensor values by using Sensor Service.

15.
a) Create an android application to get the notifications on Notification Bar by Using Notification Service.

b) Create an android application to display available Wi-Fi devices and Paired Wi-Fi devices by using Wi-Fi Service.

16.
a) Create an android application to get the Bluetooth devices and list of devices using Bluetooth and Vibrator Service.
b) Create an android application to get the System Announcements by using Broadcast Receiver.

17. Create an android application to share the data between multiple applications by using Content Provider.

18. Create an android application to display different Dialog Boxes.

19. Create an android application to display current location on Google maps by using Google-Maps Service.

References:

1. Android Application Development (with Kitkat Support), Black Book by Pradeep Kothari.
2. Beginning Android
3 Application Developments by Wei-Meng Lee.
4. Android Application Development for Dummies by Michael Burton

Laboratory Outcomes:

- Display proficiency in coding on a mobile programming platform.
- Understand the limitations and features of developing for mobile devices.
- Creating a complete Mobile app with a significant programming component, involving the sensors and hardware features of the phone.
- Practice existing state of mobile app development via researching existing apps, meeting with industry professionals, and formulating new ideas.
- Display proficiency in coding on a mobile programming platform.
- Good knowledge of economics and features of the app marketplace by offering the app for download.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Machine Learning (Lab)**	Course Code: **C131624(022)**
Total / Minimum-Pass Marks (End Semester Exam): 40/20	**L: 0 T: 0 P: 2** Credits: **1**

List of Experiments:

1. Write a python program to import and expor data using Pandas library functions
2. Demonstrate various data pre-processing techniques for a given dataset
3. Implement Dimensionality reduction using Principle Component Analysis (PCA) method.
4. Write a Python program to demonstrate various Data Visualization Techniques.
5. Implement Simple and Multiple Linear Regression Models.
6. Develop Logistic Regression Model for a given dataset.
7. Develop Decision Tree Classification model for a given dataset and use it to classify a new sample.
8. Implement Naïve Bayes Classification in Python
9. Build KNN Classification model for a given dataset.
10. Build Artificial Neural Network model with back propagation on a given dataset.
11. a) Implement Random forest ensemble method on a given dataset.
    b) Implement Boosting ensemble method on a given dataset.
12. Write a python program to implement K-Means clustering Algorithm.

Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.)

Program / Semester: **B.Tech (VI Sem)**	Branch: **Cyber Security**
Subject: **Linux Programming (Lab)**	Course Code: **C131626(022)**
Total / Minimum-Pass Marks (End Semester Exam): 40/20	**L: 0 T: 0 P: 2** Credits: **1**

List of Experiments*:*

1. Use of Basic Unix Shell Commands: ls, mkdir, rmdir, cd, cat, banner, touch, file, wc, sort, cut, grep, dd, dfspace, du, ulimit.

2. Commands related to inode, I/O redirection and piping, process control commands, mails.

3. Shell Programming: Shell script based on control structure- If-then-fi, if-thenelse-if, nested if-else, to find:
   3.1 Greatest among three numbers.
   3.2 To find a year is leap year or not.
   3.3 To input angles of a triangle and find out whether it is valid triangle or not.
   3.4 To check whether a character is alphabet, digit or special character.
   3.5 To calculate profit or loss.

4. Shell Programming - Looping- while, until, for loops
   4.1 Write a shell script to print all even and odd number from 1 to 10.
   4.2 Write a shell script to print table of a given number
   4.3 Write a shell script to calculate factorial of a given number.
   4.4 Write a shell script to print sum of all even numbers from 1 to 10.
   4.5 Write a shell script to print sum of digit of any number.

5. Shell Programming - case structure, use of break
   5.1 Write a shell script to make a basic calculator which performs addition, subtraction, Multiplication, division
   5.2 Write a shell script to print days of a week.
   5.3 Write a shell script to print starting 4 months having 31 days.

6. Shell Programming - Functions
   6.1 Write a shell script to find a number is Armstrong or not.
   6.2 Write a shell script to find a number is palindrome or not.
   6.3 Write a shell script to print Fibonacci series.
   6.4 Write a shell script to find prime number.
   6.5 Write a shell script to convert binary to decimal and decimal to binary
7. Write a shell script to print different shapes- Diamond, triangle, square, rectangle, hollow square etc.

8. Shell Programming – Arrays
   8.1 Write a C program to read and print elements of array.
   8.2 Write a C program to find sum of all array elements.
   8.3 Write a C program to find reverse of an array.
   8.4 Write a C program to search an element in an array.
   8.5 Write a C program to sort array elements in ascending or descending order