The Kelkar Education Trust's

V. G. Vaze College of Arts, Science and Commerce

(Autonomous)

Syllabus for M.Sc. – I.T.

(June 2023 Onwards)

Programme: M.Sc.

Subject : Information Technology

Semester I & II

Semester I			
Course Code	**Course Type**	**Course Title**	**Credits**
VGVPSTMDS101	Major	Data Science	4
VGVPSTMDSP101	Major	Data Science Practical	2
VGVPSTMSC101	Major	Soft Computing Techniques	4
VGVPSTMSCP101	Major	Soft Computing Techniques Practical	2
VGVPSTMCC101	Major	Cloud Computing	2
VGVPSTELP101	Elective 1 (Any One)	Security Breaches and Countermeasures Practical	4
VGVPSTEL101		Data Center Technologies	
VGVPSTEL102		Image Processing	
VGVPSTRM101	RM	Research in Computing	4
			22

Semester II			
Course Code	**Course Type**	**Course Title**	**Credits**
VGVPSTMBD201	Major	Big Data Analytics	4
VGVPSTMBDP201	Major	Big Data Analytics Practical	2
VGVPSTMMN201	Major	Modern Networking	4
VGVPSTMMNP201	Major	Modern Networking Practical	2
VGVPSTMMA201	Major	Microservices Architecture	2
VGVPSTELP201	Elective 1 (Any One)	Malware Analysis Practical	4
VGVPSTELP202		Cloud Management Practical	
VGVPSTELP203		Computer Vision Practical	
VGVPSTOJT201	OJT/FP	On job Training / Field Project	4
Total Credits :			**22**

SEMESTER I

M. Sc (Information Technology)		Semester – I		
Course Name: Data Science		**Course Code:** VGVPSTMDS101		
Periods per week (1 Period is 60 minutes)		4		
Credits		4		
			Hours	**Marks**
Evaluation System	**Theory Examination**		2	60
	Internal		--	40

Course Objective	
To enable the learner to:	
1.	Develop in depth understanding of the key technologies in data science and business analytics: data mining, machine learning, visualization techniques, predictive modeling, and statistics.
2.	Practice problem analysis and decision-making.
3.	Gain practical, hands-on experience with statistics programming languages and big data tools through coursework and applied research experiences.
4.	Understand data science methods to solve problems in real-world contexts.
5.	Analyze the need for data preprocessing and visualization techniques.

Unit	Details	Lectures
I	**Data Science Introduction & Basics** **Data Science Technology Stack:** Rapid Information Factory Ecosystem, Data Science Storage Tools, Data Lake, Data Vault, Data Warehouse Bus Matrix, Data Science Processing Tools ,Spark, Mesos, Akka , Cassandra, Kafka, Elastic Search, R ,Scala, Python, MQTT, The Future. **Layered Framework:** Definition of Data Science Framework, Cross-Industry Standard Process for Data Mining (CRISP-DM), Homogeneous Ontology for Recursive Uniform Schema, The Top Layers of a Layered Framework, Layered Framework for High-Level Data Science and Engineering **Business Layer:** Business Layer, Engineering a Practical Business Layer Utility Layer: Basic Utility Design, Engineering a Practical Utility Layer	15
II	**Statistics for Data Science**	15

	Three Management Layers: Operational Management Layer, Processing-Stream Definition and Management, Audit, Balance, and Control Layer, Balance, Control, Yoke Solution, Cause-and-Effect, Analysis System, Functional Layer, Data Science Process **Retrieve Superstep:** Data Lakes, Data Swamps, Training the Trainer Model, Understanding the Business Dynamics of the Data Lake, Actionable Business Knowledge from Data Lakes, Engineering a Practical Retrieve Superstep, Connecting to Other Data Sources. **Assess Superstep:** Assess Superstep, Errors, Analysis of Data, Practical Actions, Engineering a Practical Assess Superstep	
III	**Data Analysis with Python & Data Visualization** Process Superstep : Data Vault, Time-Person-Object-Location-Event Data Vault, Data Science Process **Transform Superstep :** Transform Superstep, Building a Data Warehouse, Transforming with Data Science, Hypothesis Testing, Overfitting and Underfitting, Precision-Recall, Cross-Validation Test.	15
IV	**Machine Learning for Data Science** **Transform Superstep:** Univariate Analysis, Bivariate Analysis, Multivariate Analysis, Linear Regression, Logistic Regression, Clustering Techniques, ANOVA, Principal Component Analysis (PCA), Decision Trees, Support Vector Machines, Networks, Clusters, and Grids, Data Mining, Pattern Recognition, Machine Learning, Bagging Data,Random Forests, Computer Vision (CV) , Natural Language Processing (NLP), Neural Networks, TensorFlow. **Organize and Report Supersteps :** Organize Superstep, Report Superstep, Graphics, Pictures, Showing the Difference	15

Course Outcome	
Learner will be able to	
CO1	Apply quantitative modeling and data analysis techniques to the solution of real world business problems, communicate findings, and effectively present results using data visualization techniques.
CO2	Apply algorithms to build machine intelligence.
CO3	Demonstrate knowledge of statistical data analysis techniques utilized in business decision making.
CO4	Apply principles of Data Science to the analysis of business problems.
CO5	Use data mining software to solve real-world problems.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Practical Data Science	Andreas François Vermeulen	APress		2018
2	Principles of Data Science	Sinan Ozdemir	PACKT		2016
3	Data Science from Scratch	Joel Grus	O'Reilly		2015
4	Data Science from Scratch first Principle in python	Joel Grus	Shroff Publishers		2017
5	Experimental Design in Data science with Least Resources	N C Das	Shroff Publishers		2018

M. Sc (Information Technology)		Semester – I	
Course Name: Data Science Practical		**Course Code:** VGVPSTMDSP101	
Periods per week (1 Period is 120 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	**100**

Course Objective
To enable the learner to:
1. Work with Cassandra.
2. Get the understanding of data processing using R.
3. Get the knowledge of error management using Pandas.
4. Understand the concept of data generation, transformation and organization.
5. carry out data visualization using power BI.

List of Practical:	
1.	Creating and using database in Cassandra

2.	Write the programs for the following:
	    i. Text Delimited CSV to HORUS format
	    ii. XML to HORUS format
	    iii. JSON to HORUS format
	    iv. MySql database to HORUS format
	    v. Picture(JPEG) to HORUS format
	    vi. Video to HORUS format
	    vii. Audio to HORUS format

3.	i. Fixers Utilities
	    ii. Data Binning or Bucketing
	    iii. Averaging of data
	    iv. Outlier Detection
	    v. Logging

4,	Perform following data processing using R
	    i. Program retrieve different attributes of data
	    ii. Data pattern
	    iii. Loading IP_DATA_ALL

5.	Perform error management on the given data using pandas package

	i. Write python/R program to create the network routing diagram from the given data on routers
	ii. Write a python/R program to build acyclic graph
	iii. Write python/R program to pick the content for BillBoards from the given data
	iv. Write a python/R program to generate GML file from given csv file
	v. Write python/R program to plan location of warehouse from the given data
	vi. Write python/R program using data science via clustering to determine new warehouse using the given data
	vii. Using the given data Write python/R program to plan the shipping routers from best-fit international logistics
	viii. Write python/R program to delete the best packing option to ship in container from
	ix. the given data
	x. Write python program to create delivery route using the given data
	xi. Write python program to crate simple forex trading planner from the given data
	xii. Write python program to process the balance sheet to ensure the only good data is processing
	xiii. Write python program to generate payroll from the given data
6.	Build the time hub, links and satellites
7.	Transforming data
8.	Organizing data
9.	Generating data
10.	Data visualisation using power Bi

Course Outcome	
Learner will be able to	
CO1	Handle Cassandra
CO2	Write a program for various conversions like video to HOURS, XML to HOURS etc.
CO3	Performs operations on data like Averaging of data, Outlier Detection, logging etc.
CO4	Write a python/R program for error management.
CO5	Use power BI for visualization.

M. Sc (Information Technology)		Semester – I	
Course Name: Soft Computing Techniques		**Course Code:** VGVPSTMSC101	
Periods per week (1 Period is 60 minutes)		4	
Credits		4	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To enable learners understand
1. Various types of soft computing techniques and applications of soft computing.
2. Concept of artificial neural network and supervised learning network.
3. Concept of unsupervised learning network, special network, third generation network.
4. Concept of fuzzy logic, fuzzy sets, fuzzification and defuzzification.
5. Fuzzy Rule base and Approximate reasoning. Genetic algorithms, Differential Evolution Algorithm.

Unit	Details	Lectures
I	**Introduction of soft computing :** soft computing vs. hard computing, various types of soft computing techniques, Fuzzy Computing, Neural Computing, Genetic Algorithms, Associative Memory, Adaptive Resonance Theory, Classification, Clustering, Bayesian Networks, Probabilistic reasoning, applications of soft computing. **Artificial Neural Network :** Fundamental concept, Evolution of Neural Networks, Basic Models, McCulloh-Pitts Neuron, Linear Separability, Hebb Network. **Supervised Learning Network :** Perceptron Networks, Adaptive Linear Neuron, Multiple Adaptive Linear Neurons, Backpropagation Network, Radial Basis Function, Time Delay Network, Functional Link Networks, Tree Neural Network	15
II	**Associative Memory Networks :** Training algorithm for pattern Association, Autoassociative memory network, hetroassociative memory network, bi-directional associative memory, Hopfield networks, iterative autoassociative memory networks, temporal associative memory networks. Kohonen self-	15

	organizing feature maps, learning vectors quantization, counter propogation networks, adaptive resonance theory networks. **Special Networks :** Simulated annealing, Boltzman machine, Gaussian Machine, Cauchy Machine, Probabilistic neural net, cascade correlation network, cognition network, neo-cognition network, cellular neural network, optical neural network **Third Generation Neural Networks :** Spiking Neural networks, convolutional neural networks, deep learning neural networks, extreme learning machine model. **UnSupervised Learning Networks :** Fixed weight competitive nets	
III	**Introduction to Fuzzy Logic, Classical Sets and Fuzzy sets :** Classical sets, Fuzzy sets. **Classical Relations and Fuzzy Relations :** Cartesian Product of relation, classical relation, fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets. **Membership Function :** features of the membership functions, fuzzification, methods of membership value assignments. **Defuzzification :** Lambda-cuts for fuzzy sets, Lambda-cuts for fuzzy relations, Defuzzification methods. **Fuzzy Arithmetic and Fuzzy measures :** fuzzy arithmetic, fuzzy measures, measures of fuzziness, fuzzy integrals.	15
IV	**Fuzzy Rule base and Approximate reasoning :** Fuzzy proportion, formation of rules, decomposition of rules, aggregation of fuzzy rules, fuzzy reasoning, fuzzy inference systems, Fuzzy logic control systems, control system design, architecture and operation of FLC system, FLC system models and applications of FLC System. **Genetic Algorithm :** Biological Background, Traditional optimization and search techniques, genetic algorithm and search space, genetic algorithm vs. traditional algorithms, basic terminologies, simple genetic algorithm, general genetic algorithm, operators in genetic algorithm, stopping condition for genetic algorithm flow, constraints in genetic algorithm, problem solving using genetic algorithm, the schema theorem, classification of genetic algorithm, Holland classifier systems, genetic programming, advantages and limitations and applications of genetic algorithm. Differential Evolution Algorithm, Hybrid soft computing techniques – neuro – fuzzy hybrid, genetic neuro-hybrid systems, genetic fuzzy hybrid and fuzzy genetic hybrid systems.	15

Course Outcome	
Learner will be able to	
CO1	Identify and describe soft computing techniques and their roles in building intelligent machines.
CO2	Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems.
CO3	Apply genetic algorithms to combinatorial optimization problems.
CO4	Apply neural networks for classification and regression problems.
CO5	Design hybrid system to revise the principles of soft computing in various applications

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Artificial Intelligence and Soft Computing	Anandita Das Battacharya	SPD	3rd	2018
2.	Principles of Soft computing	S.N.Sivanandam S.N.Deepa	Wiley	3rd	2019
3.	Neuro-Fuzzy and Soft Computing	J.S.R.Jang, C.T.Sun and E.Mizutani	Prentice Hall of India		2004
4.	Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications	S.Rajasekaran, G. A. Vijayalakshami	Prentice Hall of India		2004
5.	Fuzzy Logic with Engineering Applications	Timothy J.Ross	McGraw-Hill		1997
6.	Genetic Algorithms: Search, Optimization and Machine Learning	Davis E.Goldberg	Addison Wesley		1989
7.	Introduction to AI and Expert System	Dan W. Patterson	Prentice Hall of India		2009

M. Sc (Information Technology)	Semester – I	
Course Name: Soft Computing Technique Practical	**Course Code:** VGVPSTMSCP101	
Periods per week (1 Period is 120 minutes)	2	
Credits	2	
	Hours	**Marks**
Evaluation System **Practical Examination**	3	100

Course Objective
To enable learners understand the concept of implementation of
1.   McCulloch-Pitts neural net.
2.   Hebb's rule, delta rule, Hopfield Networ etc
3.   Back Propagation Algorithm, error Backpropagation algorithm.
4.   Kohonen Self organizing map, Linear separation, etc.
5.   Fuzzy logic, generic algorithm.

List of Practical:	
1.	**Implement the following:** Design a simple linear neural network model. Calculate the output of neural net using both binary and bipolar sigmoidal function.
2.	Implement the following: Generate AND/NOT function using McCulloch-Pitts neural net. Generate XOR function using McCulloch-Pitts neural net.
3.	Implement the Following Write a program to implement Hebb's rule. Write a program to implement of delta rule.
4,	Implement the Following Write a program for Back Propagation Algorithm Write a program for error Backpropagation algorithm.
5.	Implement the Following Write a program for Hopfield Network. Write a program for Radial Basis function
6.	Implement the Following Kohonen Self organizing map Adaptive resonance theory

7.	Implement the Following
	Write a program for Linear separation.
	Write a program for Hopfield network model for associative memory
8.	Implement the Following
	Membership and Identity Operators | in, not in,
	Membership and Identity Operators is, is not
9.	Implement the Following
	Find ratios using fuzzy logic
	Solve Tipping problem using fuzzy logic
10.	Implement the Following
	Implementation of Simple genetic algorithm
	Create two classes: City and Fitness using Genetic algorithm

Course Outcome	
Learner will be able to	
CO1	Identify and describe soft computing techniques and their roles in building intelligent machines
CO2	Recognize the feasibility of applying a soft computing methodology for a particular problem
CO3	Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems
CO4	Apply genetic algorithms to combinatorial optimization problems
CO5	Apply neural networks for classification and regression problems
CO6	Effectively use existing software tools to solve real problems using a soft computing approach
CO7	Evaluate and compare solutions by various soft computing approaches for a given problem

M. Sc (Information Technology)		Semester – I	
Course Name: Cloud Computing		**Course Code:** VGVPSTMCC101	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
Make learners
1. To learn how to use Cloud Services.
2. To implement Virtualization.
3. To implement Task Scheduling algorithms.
4. To apply Map-Reduce concept to applications.
5. To build a Private Cloud.

Unit	Details	Lectures
I	**Introduction to Cloud Computing :** Introduction, Historical developments, Building Cloud Computing Environments, **Principles of Parallel and Distributed Computing :** Eras of Computing, Parallel v/s distributed computing, Elements of Parallel Computing, Elements of distributed computing, Technologies for distributed computing. **Virtualization :** Introduction, Characteristics of virtualized environments, Taxonomy of virtualization techniques, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples. Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud usage monitor, Resource replication, Ready-made environment.	15
II	**Cloud Computing Architecture:** Introduction, Fundamental concepts and models, Roles and boundaries, Cloud Characteristics, Cloud Delivery models, Cloud Deployment models, Economics of the cloud, Open challenges. Fundamental Cloud Security: Basics, Threat agents, Cloud security threats, additional considerations. **Industrial Platforms and New Developments:** Amazon Web Services, Google App Engine, Microsoft Azure.	15

Course Outcome	
Learner will be able to	
CO1	Analyze the Cloud computing setup with its vulnerabilities and applications using different architectures.
CO2	Design different workflows according to requirements and apply map reduce programming model.
CO3	Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.
CO4	Create combinatorial auctions for cloud resources and design scheduling algorithms for computing cloud.
CO5	Assess cloud Storage systems and Cloud security, the risks involved, its impact and develop cloud application
CO6	Broadly educate to know the impact of engineering on legal and societal issues involved in addressing the security issues of cloud computing.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Mastering Cloud Computing Foundations and Applications Programming	Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi	Elsevier	-	2013
2.	Cloud Computing Concepts, Technology & Architecture	Thomas Erl, Zaigham Mahmood, and Ricardo Puttini	Prentice Hall	-	2013
3.	Distributed and Cloud Computing, From Parallel Processing to the Internet of Things	Kai Hwang, Jack Dongarra, Geoffrey Fox	MK Publishers	--	2012

M. Sc (Information Technology)	Semester – I	
Course Name: Security Breaches and Countermeasures Practical	**Course Code:** VGVPSTELP101	
Periods per week (1 Period is 120 minutes)	4	
Credits	4	
	Hours	**Marks**
Evaluation System	**Practical Examination**	3	100

Course Objective
To make learners understand and use
1. To get the insight of the security loopholes in every aspect of computing.
2. To understand the threats and different types of attacks that can be launched on computing systems.
3. To know the countermeasures that can be taken to prevent attacks on computing systems.
4. To test the software against attacks.
5. To analyse vulnerability

List of Practical:	

1.	a. Use the following tools to perform footprinting and reconnaissance
	      i. Recon-ng (Using Kali Linux)
	      ii. FOCA Tool
	      iii. Windows Command Line Utilities
	          • Ping
	          • Tracert using Ping
	          • Tracert
	          • NSLookup
	      iv. Website Copier Tool – HTTrack
	      v. Metasploit (for information gathering)
	      vi. Whois Lookup Tools for Mobile – DNS Tools, Whois, Ultra Tools Mobile
	      vii. Smart Whois
	      viii. eMailTracker Pro
	      ix. Tools for Mobile – Network Scanner, Fing – Network Tool, Network Discovery Tool, Port Droid Tool
	b. Scan the network using the following tools:

	i. Hping2 / Hping3
	ii. Advanced IP Scanner
	iii. Angry IP Scanner
	iv. Masscan
	v. NEET
	vi. CurrPorts
	vii. Colasoft Packet Builder
	viii. The Dude
2.	a. Use Proxy Workbench to see the data passing through it and save the data to file.
	b. Perform Network Discovery using the following tools:
	i. Solar Wind Network Topology Mapper
	ii. OpManager
	iii. Network View
	iv. LANState Pro
	c. Use the following censorship circumvention tools:
	i. Alkasir
	ii. Tails OS
	d. Use Scanning Tools for Mobile – Network Scanner, Fing – Network Tool, Network Discovery Tool, Port Droid Tool
3.	a. Perform Enumeration using the following tools:
	i. Nmap
	ii. NetBIOS Enumeration Tool
	iii. SuperScan Software
	iv. Hyena
	v. SoftPerfect Network Scanner Tool
	vi. OpUtils
	vii. SolarWinds Engineer's Toolset
	viii. Wireshark
	b. Perform the vulnerability analysis using the following tools:
	i. Nessus
	ii. OpenVas
4,	a. Perform mobile network scanning using NESSUS.
	b. Perform the System Hacking using the following tools:
	i. Winrtgen
	ii. PWDump
	iii. Ophcrack
	iv. Flexispy

	v. NTFS Stream Manipulation vi. ADS Spy vii. Snow viii. Quickstego ix. Clearing Audit Policies x. Clearing Logs
5.	a. Use wireshark to sniff the network. b. Use SMAC for MAC Spoofing. c. Use Caspa Network Analyser. d. Use Omnipeek Network Analyzer.

6.	a. Use Social Engineering Toolkit on Kali Linux to perform Social Engineering using Kali Linux.
	 b. Perform the DDOS attack using the following tools:
	     i. HOIC
	     ii. LOIC
	     iii. HULK
	     iv. Metasploit
	 c. Using Burp Suite to inspect and modify traffic between the browser and target application.

7.	a. Perform Web App Scanning using OWASP Zed Proxy.
	 b. Use droidsheep on mobile for session hijacking
	 c. Demonstrate the use of the following firewalls:
	     i. Zonealarm and analyse using Firewall Analyzer.
	     ii. Comodo Firewall
	 d. Use HoneyBOT to capture malicious network traffic.
	 e. Use the following tools to protect attacks on the web servers:
	     i. ID Server
	     ii. Microsoft Baseline Security Analyzer
	     iii. Syhunt Hybrid

8.	a. Protect the Web Application using dotDefender.
	 b. Demonstrate the following tools to perform SQL Injection:
	     i. Tyrant SQL
	     ii. Havij
	     iii. BBQSQL

9.	Use Aircrack-ng suite for wireless hacking and countermeasures.

10.	Use the following tools for cryptography
	     i. HashCalc
	     ii. Advanced Encryption Package

	iii. MD5 Calculator iv. TrueCrypt v. CrypTool

Course Outcome	
Learner should be able to	
CO1	Identify the different security breaches that can occur. The student should be able to evaluate the security of an organization and identify the loopholes. The student should be able to perform enumeration and network scanning.
CO2	Identify the vulnerability in the systems, breach the security of the system, identify the threats due to malware and sniff the network. The student should be able to do the penetration testing to check the vulnerability of the system towards malware and network sniffing.
CO3	Perform social engineering and educate people to be careful from attacks due to social engineering, understand and launch DoS and DDoS attacks, hijack and active session and evade IDS and Firewalls. This should help the students to make the organization understand the threats in their systems and build robust systems.
CO4	Identify the vulnerabilities in the Web Servers, Web Applications, perform SQL injection and get into the wireless networks. The student should be able to help the organization be aware about these vulnerabilities in their systems.
CO5	Identify the vulnerabilities in the newer technologies like mobiles, IoT and cloud computing. The student should be able to use different methods of cryptography.

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	CEHv10, Certified Ethical Hacker Study Guide	Ric Messier	Sybex - Wiley	-	2019
2.	All in One, Certified Ethical Hacker	Matt Walker	Tata McGraw Hill	-	2012
3.	CEH V10: EC-Council Certified Ethical Hacker Complete Training Guide	I.P. Specialist	IPSPECIALIST	-	2018

M. Sc (Information Technology)		Semester – I	
Course Name: Data Center Technologies		**Course Code:** VGVPSTEL101	
Periods per week (1 Period is 60 minutes)		4	
Credits		4	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. Identify important requirements to design and support a data center.
2. Determine a data center environment's requirements including systems and network architecture as well as services.
3. Evaluate options for server farms, network designs, high availability, load balancing, data center services, and trends that might affect data center designs.
4. Assess threats, vulnerabilities and common attacks, and network security devices available to protect data centers.
5. Design a data center infrastructure integrating features that address security, performance, and availability.
6. Measure data center traffic patterns and performance metrics.

Unit	Details	Lectures
I	Virtualization - Virtualization History and Definitions Virtualization and Network Technologies – I - Data Center Network Evolution Beginning of Network Virtualization Virtualization and Network Technologies – II - Ace Virtual Contexts Virtual Device Contexts	15
II	Fooling Spanning Tree Virtualized Chassis with Fabric Extenders - History of Data Centers Virtualization in Storage Technologies – I - Storage Evolution	15
III	Virtualization in Storage Technologies – II - Islands in SAN Secret Identities One Cable to Unite Us All Server Evolution	15
IV	Changing Personalities Transcending the Rack - Moving Targets End to End Virtualization - Virtual Data Center and Cloud Computing	15

Course Outcome	
Learner will be able to	
CO1	Understand basic concepts in Virtualization.
CO2	Use concepts of Load Balancing and Aggregation /virtual switching
CO3	Configure Data center Migration and Fabric Building
CO4	Understand various Changes in Server Architecture
CO5	Use the concepts of Cloud computing and how to move towards a cloud computing technology.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Center Virtualization Fundamentals	Gustavo Alessandro Andrade Santana	Cisco Press	1st	2014

M. Sc (Information Technology)		Semester – I	
Course Name: Image Processing		**Course Code:** VGVPSTEL102	
Periods per week (1 Period is 60 minutes)		4	
Credits		4	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. Review the fundamental concepts of a digital image processing system.
2. Analyze images in the frequency domain using various transforms.
3. Evaluate the techniques for image enhancement and image restoration.
4. Categorize various compression techniques.
5. Interpret Image compression standards.
6. Interpret image segmentation and representation techniques.

Unit	Details	Lectures
I	**Introduction:** Digital Image Processing, Origins of Digital Image Processing, Applications and Examples of Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Digital Image Fundamentals: Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Basic Relationships Between Pixels, Basic Mathematical Tools Used in Digital Image Processing, Intensity Transformations and Spatial Filtering: Basics, Basic Intensity Transformation Functions, Basic Intensity Transformation Functions, Histogram Processing, Fundamentals of Spatial Filtering, Smoothing (Lowpass) Spatial Filters, Sharpening (Highpass) Spatial Filters, Highpass, Bandreject, and Bandpass Filters from Lowpass Filters, Combining Spatial Enhancement Methods, Using Fuzzy Techniques for Intensity Transformations and Spatial Filtering	15
II	**Filtering in the Frequency Domain:** Background, Preliminary Concepts, Sampling and the Fourier Transform of Sampled Functions, The Discrete Fourier Transform of One Variable, Extensions to Functions of Two Variables, Properties of the 2-D DFT and IDFT, Basics of Filtering in the Frequency Domain, Image	15

	Smoothing Using Lowpass Frequency Domain Filters, Image Sharpening Using Highpass Filters, Selective Filtering, Fast Fourier Transform **Image Restoration and Reconstruction:** A Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only Spatial Filtering, Periodic Noise Reduction Using Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering, Constrained Least Squares Filtering, Geometric Mean Filter, Image Reconstruction from Projections Wavelet and Other Image Transforms: Preliminaries, Matrix-based Transforms, Correlation, Basis Functions in the Time-Frequency Plane, Basis Images, Fourier-Related Transforms, Walsh-Hadamard Transforms, Slant Transform, Haar Transform, Wavelet Transforms	
III	**Color Image Processing:** Color Fundamentals, Color Models, Pseudocolor Image Processing, Full-Color Image Processing, Color Transformations, Color Image Smoothing and Sharpening, Using Color in Image Segmentation, Noise in Color Images, Color Image Compression. **Image Compression and Watermarking:** Fundamentals, Huffman Coding, Golomb Coding, Arithmetic Coding, LZW Coding, Run-length Coding, Symbol-based Coding, 8 Bit-plane Coding, Block Transform Coding, Predictive Coding, Wavelet Coding, Digital Image Watermarking, Morphological Image Processing: Preliminaries, Erosion and Dilation, Opening and Closing, The Hit-or-Miss Transform, Morphological Algorithms, Morphological Reconstruction¸ Morphological Operations on Binary Images, Grayscale Morphology	15
IV	**Image Segmentation I:** Edge Detection, Thresholding, and Region Detection: Fundamentals, Thresholding, Segmentation by Region Growing and by Region Splitting and Merging, Region Segmentation Using Clustering and Superpixels, Region Segmentation Using Graph Cuts, Segmentation Using Morphological Watersheds, Use of Motion in Segmentation **Image Segmentation II:** Active Contours: Snakes and Level Sets: Background, Image Segmentation Using Snakes, Segmentation Using Level Sets. **Feature Extraction:** Background, Boundary Preprocessing, Boundary Feature Descriptors, Region Feature Descriptors, Principal Components as Feature Descriptors, Whole-Image Features, Scale-Invariant Feature Transform (SIFT)	15

Course Outcome	
Learner will be able to	
CO1	Understand the relevant aspects of digital image representation and their practical implications.
CO2	Have the ability to design pointwise intensity transformations to meet stated specifications.
CO3	Understand 2-D convolution, the 2-D DFT, and have the abitilty to design systems using these concepts.
CO4	Have a command of basic image restoration techniques.
CO5	Understand the role of alternative color spaces, and the design requirements leading to choices of color space.
CO6	Appreciate the utility of wavelet decompositions and their role in image processing systems.
CO7	Have an understanding of the underlying mechanisms of image compression, and the ability to design systems using standard algorithms to meet design specifications.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Digital Image Processing	Gonzalez and Woods	Pearson/Prentice Hall	Fourth	2018
2.	Fundamentals of Digital Image Processing	A K. Jain	PHI		
3.	The Image Processing Handbook	J. C. Russ	CRC	Fifth	2010

M. Sc (Information Technology)	Semester – I
Course Name: Research In Computing	**Course Code:** VGVPSTRM101
Periods per week (1 Period is 60 minutes)	4
Credits	4

		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
Make learners
1. Understand basics of research and the stages of the research process.
2. Get the knowledge of various data collection methods.
3. Develop the ability to explore research techniques used for solving any real world or innovate problem.
4. Able to conduct business research with an understanding of all the latest theories.

Unit	Details	Lectures
I	**Introduction:** Role of Business Research, Information Systems and Knowledge Management, Theory Building, Organization ethics and Issues **Beginning Stages of Research Process:** Problem definition, Qualitative research tools, Secondary data research	15
II	**Research Methods and Data Collection:** Survey research, communicating with respondents, Observation methods, Experimental research	15
III	**Measurement Concepts, Sampling and Field work:** Levels of Scale measurement, attitude measurement, questionnaire design, sampling designs and procedures, determination of sample size	15
IV	**Data Analysis and Presentation:** Editing and Coding, Basic Data Analysis, Univariate Statistical Analysis and Bivariate Statistical analysis and differences between two variables. Multivariate Statistical Analysis.	15

Course Outcome	
Learner should be able to	
CO1	Solve real world problems with a scientific approach.
CO2	Develop analytical skills by applying scientific methods.
CO3	Recognize, understand and apply the language, theory and models of the field of business analytics
CO4	Foster an ability to critically analyze, synthesize and solve complex unstructured business problems
CO5	Understand and critically apply the concepts and methods of business analytics.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Business Research Methods	William G.Zikmund, B.J Babin, J.C. Carr, Atanu Adhikari, M.Griffin	Cengage	8e	2016
2.	Business Analytics	Albright Winston	Cengage	5e	2015
3.	Research Methods for Business Students Fifth Edition	Mark Saunders			2011
4.	Multivariate Data Analysis	Hair	Pearson	7e	2014

SEMESTER II

M. Sc (Information Technology)		Semester – II	
Course Name: Big Data Analytics		**Course Code:** VGVPSTMBD201	
Periods per week (1 Period is 60 minutes)		**4**	
Credits		**4**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. To provide an overview of an exciting growing field of big data analytics.
2. To introduce the tools required to manage and analyze big data like Hadoop, NoSql MapReduce.
3. To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
4. To enable students to have skills that will help them to solve complex real-world problems in for decision support

Unit	Details	Lectures
I	**Introduction :**Introduction to Big Data, Characteristics of Data, and Big Data Evolution of Big Data, Definition of Big Data, Challenges with big data, Why Big data? Data Warehouse environment, Traditional Business Intelligence versus Big Data. State of Practice in Analytics, Key roles for New Big Data Ecosystems, Examples of big Data Analytics. Big Data Analytics, Introduction to big data analytics, Classification of Analytics, Challenges of Big Data, Importance of Big Data, Big Data Technologies, Data Science, Responsibilities, Soft state eventual consistency. Data Analytics Life Cycle	15
II	**Analytical Theory and Methods:** Clustering and Associated Algorithms, Association Rules, Apriori Algorithm, Candidate Rules, Applications of Association Rules, Validation and Testing, Diagnostics, Regression, Linear Regression, Logistic Regression, Additional Regression Models. **Analytical Theory and Methods:** Classification, Decision Trees, Naïve Bayes, Diagnostics of Classifiers, Additional Classification Methods, Time Series Analysis, Box Jenkins methodology, ARIMA Model, Additional methods. Text Analysis, Steps, Text Analysi s Example, Collecting Raw Text,	15

		Representing Text, Term Frequency-Inverse Document Frequency (TFIDF), Categorizing Documents by Topics, Determining Sentiments	
III		Data Product, Building Data Products at Scale with Hadoop, Data Science Pipeline and Hadoop Ecosystem, Operating System for Big Data, Concepts, Hadoop Architecture, Working with Distributed file system, Working with Distributed Computation, Framework for Python and Hadoop Streaming, Hadoop Streaming, MapReduce with Python, Advanced MapReduce. In-Memory Computing with Spark, Spark Basics, Interactive Spark with PySpark, Writing Spark Applications	15
IV		Distributed Analysis and Patterns, Computing with Keys, Design Patterns, Last- Mile Analytics, Data Mining and Warehousing, Structured Data Queries with Hive, HBase, Data Ingestion, Importing Relational data with Sqoop, Injesting stream data with flume. Analytics with higher level APIs, Pig, Spark's higher level APIs.	15

Course Outcome	
Learner will be able to	
CO1	Understand the characteristics of Big Data and its related concepts.
CO2	Understand the **a**nalytical theory and its methods, Classification methods
CO3	Familiarize with Hadoop Architecture, Hadoop Ecosystem, and Working with Distributed file systems.
CO4	Know the softwares available and used for Big Data Analysis
CO5	Make a choice of tools(Hadoop, NoSql MapReduce) for analysis of Big Data

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Big Data and Analytics	Subhashini Chellappan Seema Acharya	Wiley	First	
2.	Data Analytics with Hadoop *An Introduction for Data Scientists*	*Benjamin Bengfort and Jenny Kim*	O'Reilly		2016
3.	Big Data and Hadoop	V.K Jain	Khanna Publishing	First	2018

M. Sc (Information Technology)		Semester – II	
Course Name: Big Data Analytics Practical		**Course Code:** VGVPSTMBDP201	
Periods per week (1 Period is 120 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	100

Course Objective
To make learners understand
1. The installation of Hadoop and HDFS.
2. get the knowledge of implementation of MapReduce for word count/ frequency, processing of weather dataset.
3. The configuration of Hive and applications in Hive.
4. Implement classification techniques like Decision tree, SVM .
5. the working of Regression Model, Multiple Regression Model, Classification Model, Clustering Model

List of Practical:	
1.	Install, configure and run Hadoop and HDFS and explore HDFS.
2.	Implement word count / frequency programs using MapReduce
3.	Implement MapReduce program that processes a weather dataset.
4,	Implement an application that stores big data in Hbase / MongoDB and manipulate it using R / Python
5.	Implement the program in practical 4 using Pig.
6.	Configure the Hive and implement the application in Hive.
7.	Write a program to illustrate the working of Jaql.
8.	Implement the following: a. Implement Decision tree classification techniques b. Implement SVM classification techniques
9.	Solve the following: **a. REGRESSION MODEL** Import data from web storage. Name the dataset and now do Logistic Regression to find out relation between variables that are affecting the admission of a student in an institute based on his or her GRE score, GPA obtained and rank of the student. Also check the model is fit or not. require (foreign), require(MASS).

	b. MULTIPLE REGRESSION MODEL Apply multiple regressions, if data have a continuous independent variable. Apply on above dataset.

10.	Solve the Following:
	 a. CLASSIFICATION MODEL
	     i. Install relevant package for classification.
	     ii. Choose classifier for classification problem.
	     iii. Evaluate the performance of classifier.
	 b. CLUSTERING MODEL
	     i. Clustering algorithms for unsupervised classification.
	     ii. b. Plot the cluster data using R visualizations.

Course Outcome	
Learner will be able to	
CO1	Install, configure and run Hadoop, HDFS and explore HDFS
CO2	Understand the key issues in big data management and its associated applications in intelligent business and scientific computing.
CO3	Acquire fundamental enabling techniques and scalable algorithms like Hadoop, Map Reduce and NO SQL in big data analytics.
CO4	Interpret business models and scientific computing paradigms, and apply software tools for big data analytics.
CO5	Achieve adequate perspectives of big data analytics in various applications like recommender systems, social media applications etc

M. Sc (Information Technology)		Semester – II	
Course Name: Modern Networking		**Course Code:** VGVPSTMMN201	
Periods per week (1 Period is 60 minutes)		4	
Credits		4	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. To understand the state-of-the-art in network protocols, architectures and applications.
2. Analyze existing network protocols and networks.
3. Develop new protocols in networking
4. To understand how networking research is done
5. To investigate novel ideas in the area of Networking via term-long research projects

Unit	Details	Lectures
I	Modern Networking **Elements of Modern Networking :** The Networking Ecosystem ,Example Network Architectures,Global Network Architecture,A Typical Network Hierarchy Ethernet Applications of Ethernet Standards Ethernet Data Rates Wi-Fi Applications of Wi-Fi,Standards Wi-Fi Data Rates 4G/5G Cellular First Generation Second Generation, Third Generation Fourth Generation Fifth Generation, Cloud Computing Cloud Computing Concepts The Benefits of Cloud Computing Cloud Networking Cloud Storage, Internet of Things Things on the Internet of Things, Evolution Layers of the Internet of Things, Network Convergence Unified Communications, Requirements and Technology Types of Network and Internet Traffic, Elastic Traffic, Inelastic Traffic, Real-Time Traffic Characteristics Demand: Big Data, Cloud Computing, and Mobile TrafficBig Data Cloud Computing, Mobile Traffic, Requirements: QoS and QoE,,Quality of Service, Quality of Experience, Routing Characteristics, Packet Forwarding, Congestion Control ,Effects of Congestion, Congestion Control Techniques, SDN and NFV Software-Defined Networking, Network Functions Virtualization Modern Networking Elements	15
II	**Software-Defined Networks SDN:** Background and Motivation, Evolving	15

	Network Requirements Demand Is Increasing,Supply Is IncreasingTraffic Patterns Are More ComplexTraditional Network Architectures are Inadequate, The SDN Approach Requirements SDN Architecture Characteristics of Software-Defined Networking, SDN- and NFV-Related Standards Standards-Developing Organizations Industry Consortia Open Development Initiatives, SDN Data Plane and OpenFlow SDN Data Plane, Data Plane Functions Data Plane Protocols OpenFlow Logical Network Device Flow Table Structure Flow Table Pipeline, The Use of Multiple Tables Group Table OpenFlow Protocol, SDN Control Plane SDN Control Plane Architecture Control Plane Functions, Southbound Interface Northbound InterfaceRouting, ITU-T Model, OpenDaylight OpenDaylight Architecture OpenDaylight Helium, REST REST Constraints Example REST API, Cooperation and Coordination Among Controllers, Centralized Versus Distributed Controllers, High-Availability Clusters Federated SDN Networks, Border Gateway Protocol Routing and QoS Between Domains, Using BGP for QoS Management IETF SDNi OpenDaylight SNDi SDN Application Plane SDN Application Plane Architecture Northbound Interface Network Services Abstraction Layer Network Applications, User Interface, Network Services Abstraction Layer Abstractions in SDN, Frenetic Traffic Engineering PolicyCop Measurement and Monitoring Security OpenDaylight DDoS Application Data Center Networking, Big Data over SDN Cloud Networking over SDN Mobility and Wireless Information-Centric Networking CCNx, Use of an Abstraction Layer	
III	Virtualization, Network Functions Virtualization: Concepts and Architecture, Background and Motivation for NFV, Virtual Machines The Virtual Machine Monitor, Architectural Approaches Container Virtualization, NFV Concepts Simple Example of the Use of NFV, NFV Principles High-Level NFV Framework, NFV Benefits and Requirements NFV Benefits, NFV Requirements, NFV Reference Architecture NFV Management and Orchestration, Reference Points Implementation, NFV Functionality, NFV Infrastructure, Container Interface, Deployment of NFVI Containers, Logical Structure of NFVI Domains, Compute Domain, Hypervisor Domain, Infrastructure Network Domain, Virtualized Network Functions, VNF Interfaces, VNFC to VNFC Communication, VNF Scaling, NFV Management and Orchestration, Virtualized Infrastructure Manager, Virtual Network Function Manager, NFV Orchestrator, Repositories, Element Management, OSS/BSS, NFV Use Cases Architectural Use Cases, Service-Oriented Use Cases, SDN and NFV Network Virtualization, Virtual LANs ,The Use of Virtual LANs,Defining	15

	VLANs, Communicating VLAN Membership,IEEE 802.1Q VLAN Standard, Nested VLANs, OpenFlow VLAN Support, Virtual Private Networks, IPsec VPNs,MPLS VPNs, Network Virtualization, Simplified Example, Network Virtualization Architecture, Benefits of Network Virtualization, OpenDaylight's Virtual Tenant Network, Software-Defined Infrastructure,Software-Defined Storage, SDI Architecture	
IV	Defining and Supporting User Needs, Quality of Service, Background, QoS Architectural Framework, Data Plane, Control Plane, Management Plane, Integrated Services Architecture, ISA Approach ISA Components, ISA Services, Queuing Discipline, Differentiated Services, Services, DiffServ Field, DiffServ Configuration and Operation, Per-Hop Behavior, Default Forwarding PHB, Service Level Agreements, IP Performance Metrics, OpenFlow QoS Support, Queue Structures, Meters, QoE: User Quality of Experience, Why QoE?,Online Video Content Delivery, Service Failures Due to Inadequate QoE Considerations QoE-Related Standardization Projects, Definition of Quality of Experience, Definition of Quality, Definition of Experience Quality Formation Process, Definition of Quality of Experience, QoE Strategies in Practice, The QoE/QoS Layered Model Summarizing and Merging the ,QoE/QoS Layers, Factors Influencing QoE, Measurements of QoE, Subjective Assessment, Objective Assessment, End-User Device Analytics, Summarizing the QoE Measurement Methods, Applications of QoE Network Design Implications of QoS and QoE Classification of QoE/ QoS Mapping Models, Black-Box Media-Based QoS/QoE Mapping Models, Glass- Box Parameter-Based QoS/QoE Mapping Models,Gray-Box QoS/QoE Mapping Models, Tips for QoS/QoE Mapping Model Selection,IP-Oriented Parameter- Based QoS/QoE Mapping Models,Network Layer QoE/QoS Mapping Models for Video Services, Application Layer QoE/QoS Mapping Models for Video Services Actionable QoE over IP-Based Networks, The System-Oriented Actionable QoE Solution, The Service-Oriented Actionable QoE Solution, QoE Versus QoS Service Monitoring, QoS Monitoring Solutions, QoE Monitoring Solutions, QoE-Based Network and Service Management, QoE-Based Management of VoIP Calls, QoE-Based Host-Centric Vertical Handover, QoE-Based Network-Centric Vertical Handover	15

Course Outcome	
Learner will be able to	
CO1	Demonstrate in-depth knowledge in the area of Computer Networking
CO2	Demonstrate scholarship of knowledge through performing in a group to identify, formulate and solve a problem related to Computer Networks
CO3	Prepare a technical document for the identified Networking System
CO4	Conducting experiments to analyze the identified research work in building Computer Networks
CO5	Understand the research in networking.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud	William Stallings	Addison-Wesley Professional		October 2015
2.	SDN and NFV Simplified A Visual Guide to Understanding Software Defined Networks and Network Function Virtualization	Jim Doherty	Pearson Education, Inc		
3.	Network Functions Virtualization (NFV) with a Touch of SDN	Rajendra Chayapathi Syed Farrukh Hassan	Addison-Wesley		
4.	CCIE and CCDE Evolving Technologies Study Guide	Brad dgeworth, Jason Gooley, Ramiro Garza Rios	Pearson Education, Inc		2019

M. Sc (Information Technology)		Semester – II	
Course Name: Modern Networking Practical		**Course Code:** VGVPSTMMNP201	
Periods per week (1 Period is 120 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	**100**

Course Objective
To make learners understand
1. Concept of SLA Tracking, Path Control Topology, AS_PATH attribute etc.
2. Configuration of iBGP, EBGP.
3. Configuration of PBR, Inter-VLAN Routing.
4. Concept of IP Service Level Agreements.
5. Simulation of SDN, OpenFlow, MPLS..

List of Practical:	
1.	Configure IP SLA Tracking and Path Control Topology
2.	Using the AS_PATH Attribute
3.	Configuring IBGP and EBGP Sessions, Local Preference, and MED
4,	Secure the Management Plane
5.	Configure and Verify Path Control Using PBR
6.	IP Service Level Agreements and Remote SPAN in a Campus Environment
7.	Inter-VLAN Routing
8.	Simulating MPLS environment and Simulating VRF
9.	Simulating SDN with - OpenDaylight SDN Controller with the Mininet Network Emulator - OFNet SDN network emulator
10.	Simulating OpenFlow Using MININET

Course Outcome	
Learner will be able to	
CO1	Demonstrate in-depth knowledge in the area of Computer Networking.
CO2	Configure IP SLA, Path Control Topology, AS_PATH attribute etc.
CO3	Configure iBGP, EBGP, Inter-VLAN routing
CO4	Simulate SDN.
CO5	Understand and Simulate OPenFlow, SDN, MPLS.

M. Sc (Information Technology)		Semester – II	
Course Name: Microservice Architecture		**Course Code:** VGVPSTMMA201	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. Gain a thorough understanding of the philosophy and architecture of Web applications using ASP.NET Core MVC
2. Gain a practical understanding of .NET Core.
3. Acquire a working knowledge of Web application development using ASP.NET Core MVC 6 and Visual Studio
4. Persist data with XML Serialization and ADO.NET with SQL Server Create HTTP services using ASP.NET Core Web API.
5. Deploy ASP.NET Core MVC applications to the Windows Azure cloud

Unit	Details	Lectures
I	**Microservices:** Understanding Microservices, Adopting Microservices, The Microservices Way. **Microservices Value Proposition:** Deriving Business Value, defining a Goal-Oriented, Layered Approach, Applying the Goal-Oriented, Layered Approach. **Designing Microservice Systems:** The Systems Approach to Microservices, A Microservices Design Process, Establishing a Foundation Goals and Principles, Platforms, Culture.	15
II	**Service Design:** Microservice Boundaries, API design for Microservices, Data and Microservices, Distributed Transactions and Sagas, Asynchronous Message- Passing and Microservices, dealing with Dependencies, **System Design and Operations:** Independent Deployability, More Servers, Docker and Microservices, Role of Service Discovery, Need for an API Gateway, Monitoring and Alerting. **Adopting Microservices in Practice:** Solution Architecture Guidance, Organizational Guidance, Culture Guidance, Tools and Process Guidance, Services Guidance.	15

Course Outcome	
Learner will be able to	
CO1	Develop web applications using Model View Controller.
CO2	Think and apply the microservices way to software development.
CO3	Gaining a thorough understanding of the philosophy and architecture of .NET Core
CO4	Acquiring a working knowledge of the .NET programming model.
CO5	Implementing multi-threading effectively in .NET applications.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Microservice Architecture: *Aligning Principles, Practices, and Culture*	Irakli Nadareishvili, Ronnie Mitra, Matt McLarty, and Mike Amundsen	O'Reilly	First	2016
2.	Building Microservices with ASP.NET Core	Kevin Hoffman	O'Reilly	First	2017
3.	Building Microservices: Designing Fine-Grained Systems	Sam Newman	O'Reilly	First	
4.	Production-ready Microservices	Susan J. Fowler	O'Reilly		2016

The Kelkar Education Trust's
V G Vaze College of Arts, Science and Commerce
(Autonomous)

M. Sc (Information Technology)		Semester – II	
Course Name: Malware Analysis Practical		**Course Code:** VGVPSTELP201	
Periods per week (1 Period is 120 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	100

Course Objective
To make learners understand and use
1. To understand the type of malware and its functionality.
2. Determine how the system was infected by malware and define if it was a targeted attack or a phishing attack.
3. How malware communicates with an attacker.   4.Future detection of malware and generating signatures.
5. Investigate, detect, and respond to various types of malware threat.

List of Practical:	
1.	**a. Files: *Lab01-01.exe* and *Lab01-01.dll*.**
	i. Upload the files to *http://www.VirusTotal.com/* and view the reports. Does either file match any existing antivirus signatures?
	ii. When were these files compiled?
	iii. Are there any indications that either of these files is packed or obfuscated? If so, what are these indicators?
	iv. Do any imports hint at what this malware does? If so, which imports are they?
	v. Are there any other files or host-based indicators that you could look for on infected systems?
	vi. What network-based indicators could be used to find this malware on infected machines?
	vii. What would you guess is the purpose of these files?
	b. Analyze the file *Lab01-02.exe*.
	i. Upload the *Lab01-02.exe* file to *http://www.VirusTotal.com/*. Does it match any existing antivirus definitions?
	ii. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.
	iii. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

	iv. What host- or network-based indicators could be used to identify this malware on infected machines?
	c. Analyze the file Lab01-03.exe.
	i. Upload the *Lab01-03.exe* file to *http://www.VirusTotal.com/*. Does it match any existing antivirus definitions?
	ii. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.
	iii. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?
	iv. What host- or network-based indicators could be used to identify this malware on infected machines?
	d. Analyze the file Lab01-04.exe.
	i. Upload the *Lab01-04.exe* file to *http://www.VirusTotal.com/*. Does it match any existing antivirus definitions?
	ii. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.
	iii. When was this program compiled?
	iv. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?
	v. What host- or network-based indicators could be used to identify this malware on infected machines?
	vi. This file has one resource in the resource section. Use Resource Hacker to examine that resource, and then use it to extract the resource. What can you learn from the resource?
	e. Analyze the malware found in the file Lab03-01.exe using basic dynamic analysis tools.
	i. What are this malware's imports and strings?
	ii. What are the malware's host-based indicators?
	iii. Are there any useful network-based signatures for this malware? If so, what are they?
	f. Analyze the malware found in the file Lab03-02.dll using basic dynamic analysis tools.
	i. How can you get this malware to install itself?
	ii. How would you get this malware to run after installation?
	iii. How can you find the process under which this malware is running?
	iv. Which filters could you set in order to use procmon to glean information?
	v. What are the malware's host-based indicators?
	vi. Are there any useful network-based signatures for this malware?
	g. Execute the malware found in the file Lab03-03.exe while monitoring it using basic dynamic analysis tools in a safe environment

2	i.	What do you notice when monitoring this malware with Process Explorer?
	ii.	Can you identify any live memory modifications?
	iii.	What are the malware's host-based indicators?
	iv.	What is the purpose of this program?
	h. Analyze the malware found in the file Lab03-04.exe using basic dynamic analysis tools.	
	i.	What happens when you run this file?
	ii.	What is causing the roadblock in dynamic analysis?
	iii.	Are there other ways to run this program?
2.	**a. Analyze the malware found in the file Lab05-01.dll using only IDA Pro. The goal of this lab is to give you hands-on experience with IDA Pro. If you've already worked with IDA Pro, you may choose to ignore these questions and focus on reverse-engineering the malware.**	
	i.	What is the address of DllMain?
	ii.	Use the Imports window to browse to gethostbyname. Where is the import located?
	iii.	How many functions call gethostbyname?
	iv.	Focusing on the call to gethostbynamelocated at 0x10001757, can you fig- ure out which DNS request will be made?
	v.	How many local variables has IDA Pro recognized for the subroutine at 0x10001656?
	vi.	How many parameters has IDA Pro recognized for the subroutine at 0x10001656?
	vii.	Use the Strings window to locate the string \cmd.exe /cin the disassembly. Where is it located?
	viii.	What is happening in the area of code that references \cmd.exe/c?
	ix.	In the same area, at 0x100101C8, it looks like dword_1008E5C4 is a global variable that helps decide which path to take. How does the malware set dword_1008E5C4? (Hint: Use dword_1008E5C4's cross-references.)
	x.	A few hundred lines into the subroutine at 0x1000FF58, a series of com- parisons use memcmp to compare strings. What happens if the string compar- ison to robotwork is successful (when memcmp returns 0)?
	xi.	What does the export PSLISTdo?
	xii.	Use the graph mode to graph the cross-references from sub_10004E79. Which API functions could be called by entering this function? Based on the API functions alone, what could you rename this function?
	xiii.	How many Windows API functions does DllMaincall directly? How many at a depth of 2?

	xiv.	At 0x10001358, there is a call to Sleep (an API function that takes one parameter containing the number of milliseconds to sleep). Looking backward through the code, how long will the program sleep if this code executes?
	xv.	At 0x10001701 is a call to socket. What are the three parameters?
	xvi.	Using the MSDN page for socketand the named symbolic constants func- tionality in IDA Pro, can you make the parameters more meaningful? What are the parameters after you applychanges?
	xvii.	Search for usage of the in instruction (opcode 0xED). This instruction is used with a magic string VMXh to perform VMware detection. Is that in use in this malware? Using the cross - references to the function that executes the in instruction, is there further evidence of VMware detection?
	xviii.	Jump your cursor to 0x1001D988. What do you find?
	xix.	If you have the IDA Python plug-in installed (included with the com- mercial version of IDA Pro), run *Lab05-01.py*, an IDA Pro Python script provided with the malware for this book. (Make sure the cursor is at 0x1001D988.) What happens after you run the script?
	xx.	With the cursor in the same location, how do you turn this data into a single ASCII string?
	xxi.	Open the script with a text editor. How does it work?
b.	**analyze the malware found in the file Lab06-01.exe.**	
	i.	What is the major code construct found in the only subroutine called by main?
	ii.	What is the subroutine located at 0x40105F?
	iii.	What is the purpose of this program?
c.	**Analyze the malware found in the file Lab06-02.exe.**	
	i.	What operation does the first subroutine called by mainperform?
	ii.	What is the subroutine located at 0x40117F?
	iii.	What does the second subroutine called by maindo?
	iv.	What type of code construct is used in this subroutine?
	v.	Are there any network-based indicators for this program?
	vi.	What is the purpose of this malware?
d.	**analyze the malware found in the file Lab06-03.exe.**	
	i.	Compare the calls in mainto Lab 6-2's mainmethod. What is the new function called from main?
	ii.	What parameters does this new function take?
	iii.	What major code construct does this function contain?
	iv.	What can this function do?
	v.	Are there any host-based indicators for this malware?
	vi.	What is the purpose of this malware?

	e. analyze the malware found in the file Lab06-04.exe.
	i. What is the difference between the calls made from the main method in Labs 6-3 and 6-4?
	ii. What new code construct has been added to main?
	iii. What is the difference between this lab's parse HTML function and those of the previous labs?
	iv. How long will this program run? (Assume that it is connected to the Internet.)
	v. Are there any new network-based indicators for this malware?
	vi. What is the purpose of this malware?
3.	**a. Analyze the malware found in the file Lab07-01.exe.**
	i. How does this program ensure that it continues running (achieves per- sistence) when the computer is restarted?
	ii. Why does this program use a mutex?
	iii. What is a good host-based signature to use for detecting this program?
	iv. What is a good network-based signature for detecting this malware?
	v. What is the purpose of this program?
	vi. When will this program finish executing?
	b. Analyze the malware found in the file Lab07-02.exe.
	i. How does this program achieve persistence?
	ii. What is the purpose of this program?
	iii. When will this program finish executing?
	c. For this lab, we obtained the malicious executable, Lab07-03.exe, and DLL, Lab07-03.dll, prior to executing. This is important to note because the mal- ware might change once it runs. Both files were found in the same directory on the victim machine. If you run the program, you should ensure that both files are in the same directory on the analysis machine. A visible IP string beginning with 127 (a loopback address) connects to the local machine. (In the real version of this malware, this address connects to a remote machine, but we've set it to connect to localhost to protect you.)
	i. How does this program achieve persistence to ensure that it continues running when the computer is restarted?
	ii. What are two good host-based signatures for this malware?
	iii. What is the purpose of this program?
	iv. How could you remove this malware once it is installed?
	d. Analyze the malware found in the file Lab09-01.exe using OllyDbg and IDA Pro to answer the following questions. This malware was initially analyzed in the Chapter 3 labs using basic static and dynamic analysis techniques.
	i. How can you get this malware to install itself?

	ii.	What are the command-line options for this program? What is the pass- word requirement?
	iii.	How can you use OllyDbg to permanently patch this malware, so that it doesn't require the special command-line password?
	iv.	What are the host-based indicators of this malware?
	v.	What are the different actions this malware can be instructed to take via the network?
	vi.	Are there any useful network-based signatures for this malware?
	e.	**Analyze the malware found in the file Lab09-02.exe using OllyDbg to answer the following questions.**
	i.	What strings do you see statically in the binary?
	ii.	What happens when you run this binary?
	iii.	How can you get this sample to run its malicious payload?
	iv.	What is happening at 0x00401133?
	v.	What arguments are being passed to subroutine 0x00401089?
	vi.	What domain name does this malware use?
	vii.	What encoding routine is being used to obfuscate the domain name?
	viii.	What is the significance of the CreateProcessAcall at 0x0040106E?
	f.	**Analyze the malware found in the file Lab09-03.exe using OllyDbg and IDA Pro. This malware loads three included DLLs (DLL1.dll, DLL2.dll, and DLL3.dll) that are all built to request the same memory load location. Therefore, when viewing these DLLs in OllyDbg versus IDA Pro, code may appear at different memory locations. The purpose of this lab is to make you comfortable with finding the correct location of code within IDA Pro when you are looking at code in OllyDbg**
	i.	What DLLs are imported by *Lab09-03.exe*?
	ii.	What is the base address requested by *DLL1.dll*, *DLL2.dll*, and *DLL3.dll*?
	iii.	When you use OllyDbg to debug *Lab09-03.exe*, what is the assigned based address for: *DLL1.dll*, *DLL2.dll*, and *DLL3.dll*?
	iv.	When *Lab09-03.exe* calls animport functionfrom *DLL1.dll*, whatdoes this import function do?
	v.	When *Lab09-03.exe* calls WriteFile, what is the filename it writes to?
	vi.	vi. When *Lab09-03.exe* creates a job using NetScheduleJobAdd, where does it get the data for the second parameter?
	vii.	While running or debugging the program, you will see that it prints out three pieces of mystery data. What are the following: DLL 1 mystery data 1, DLL 2 mystery data 2, and DLL 3 mystery data 3?
	viii.	How can you load *DLL2.dll* into IDA Pro so that it matches the load address used by OllyDbg?

4.	**a. This lab includes both a driver and an executable. You can run the executable from anywhere, but in order for the program to work properly, the driver must be placed in the C:\Windows\ System32 directory where it was origi- nally found on the victim computer. The executable is Lab10-01.exe, and the driver is Lab10-01.sys.**
	i. Does this program make any direct changes to the registry? (Use procmon to check.)
	ii. The user-space program calls the ControlService function. Can you set a breakpoint with WinDbg to see what is executed in the kernel as a result of the call to ControlService?
	iii. What does this program do?
	b. The file for this lab is Lab10-02.exe.
	i. Does this program create any files? If so, what are they?
	ii. Does this program have a kernel component?
	iii. What does this program do?
	c. This lab includes a driver and an executable. You can run the executable from anywhere, but in order for the program to work properly, the driver must be placed in the C:\Windows\System32 directory where it was originally found on the victim computer. The executable is Lab10-03.exe, and the driver is Lab10-03.sys.
	i. What does this program do?
	ii. Once this program is running, how do you stop it?
	iii. What does the kernel component do?
5.	**a. Analyze the malware found in Lab11-01.exe**
	i. What does the malware drop to disk?
	ii. How does the malware achieve persistence?
	iii. How does the malware steal user credentials?
	iv. What does the malware do with stolen credentials?
	v. How can you use this malware to get user credentials from your test environment?
	b. Analyze the malware found in *Lab11-02.dll*. Assume that a suspicious file named *Lab11-02.ini* was also found with this malware.
	i. What are the exports for this DLL malware?
	ii. What happens after you attempt to install this malware using *rundll32.exe*?
	iii. Where must *Lab11-02.ini* reside in order for the malware to install properly?
	iv. How is this malware installed for persistence?
	v. What user-space rootkit technique does this malware employ?
	vi. What does the hooking code do?
	vii. Which process(es) does this malware attack and why?

viii.	What is the significance of the *.ini* file?
	c. Analyze the malware foundin *Lab11-03.exe* and *Lab11-03.dll*. Makesurethat both files are in the same directory duringanalysis
i.	What interesting analysis leads can you discover using basic static analysis?
ii.	What happens when you run this malware?
iii.	How does *Lab11-03.exe* persistently install *Lab11-03.dll*?
iv.	Which Windows system file does the malware infect?
v.	What does *Lab11-03.dll* do?
vi.	Where does the malware store the data it collects?
6. a.	**Analyze the malware found in the file *Lab12-01.exe* and *Lab12-01.dll*. Make sure that these files are in the same directory when performing the analysis.**
i.	What happens when you run the malware executable?
ii.	What process is being injected?
iii.	How can you make the malware stop the pop-ups?
iv.	How does this malware operate?
	b. Analyze the malware found in the file *Lab12-02.exe*.
i.	What is the purpose of this program?
ii.	How does the launcher program hide execution?
iii.	Where is the malicious payload stored?
iv.	How is the malicious payload protected?
v.	How are strings protected?
	c. Analyze the malware extracted during the analysis of Lab 12-2, or use the file *Lab12-03.exe*.
i.	What is the purpose of this malicious payload?
ii.	How does the malicious payload inject itself?
iii.	What filesystem residue does this program create?
	d. Analyze the malware found in the file *Lab12-04.exe*.
i.	What does the code at 0x401000 accomplish?
ii.	Which process has code injected?
iii.	What DLL is loaded using LoadLibraryA?
iv.	What is the fourth argument passed to the CreateRemoteThread call?
v.	What malware is dropped by the main executable?
7. a.	**Analyze the malware found in the file *Lab13-01.exe*.**
i.	Compare the strings in the malware (from the output of the stringscommand) with the information available via dynamic analysis. Based on this comparison, which elements might be encoded?
ii.	Use IDA Pro to look for potential encoding by searching for the string xor. What type of

		encoding do you find?
	iii.	What is the key used for encoding and what content does it encode?
	iv.	Use the static tools FindCrypt2, Krypto ANALyzer (KANAL), and the IDA Entropy Plugin to identify any other encoding mechanisms. What do you find?
	v.	What type of encoding is used for a portion of the network traffic sent by the malware?
	vi.	Where is the Base64 function in the disassembly?
	vii.	What is the maximum length of the Base64-encoded data that is sent? What is encoded?
	viii.	In this malware, would you ever see the padding characters (=or ==) in the Base64-encoded data?
	ix.	What does this malware do?
	b.	**Analyze the malware found in the file *Lab13-02.exe*.**
	i.	Using dynamic analysis, determine what this malware creates.
	ii.	Use static techniques such as an xor search, FindCrypt2, KANAL, and the IDA Entropy Plugin to look for potential encoding. What do you find?
	iii.	Based on your answer to question 1, which imported function would be a good prospect for finding the encoding functions?
	iv.	Where is the encoding function in the disassembly?
	v.	Trace from the encoding function to the source of the encoded content. What is the content?
	vi.	Can you find the algorithm used for encoding? If not, how can you decode the content?
	vii.	Using instrumentation, can you recover the original source of one of the encoded files?
	c.	**Analyze the malware found in the file *Lab13-03.exe*.**
	i.	Compare the output of strings with the information available via dynamic analysis. Based on this comparison, which elements might be encoded?
	ii.	Use static analysis to look for potential encoding by searching for the string xor. What type of encoding do you find?
	iii.	Use static tools like FindCrypt2, KANAL, and the IDA Entropy Plugin to identify any other encoding mechanisms. How do these findings com- pare with the XOR findings?
	iv.	Which two encoding techniques are used in this malware?
	v.	For each encoding technique, what is the key?
	vi.	For the cryptographic encryption algorithm, is the key sufficient? What else must be known?
	vii.	What does this malware do?
	viii.	Create code to decrypt some of the content produced during dynamic analysis. What is this content?
8.	**a.**	**Analyze the malware found in file *Lab14-01.exe*. This program is not harmful to your system.**

i.	Which networking libraries does the malware use, and what are their advantages?
ii.	What source elements are used to construct the networking beacon, and what conditions would cause the beacon to change?
iii.	Why might the information embedded in the networking beacon be of interest to the attacker?
iv.	Does the malware use standard Base64 encoding? If not, how is the encoding unusual?
v.	What is the overall purpose of this malware?
vi.	What elements of the malware's communication may be effectively detected using a network signature?
vii.	What mistakes might analysts make in trying to develop a signature for this malware?
viii.	What set of signatures would detect this malware (and future variants)?
	b. Analyze the malware found in file *Lab14-02.exe*. This malware has been configured to beacon to a hard-coded loopback address in order to prevent it from harming your system, but imagine that it is a hard-coded external address.
i.	What are the advantages or disadvantages of coding malware to use direct IP addresses?
ii.	Which networking libraries does this malware use? What are the advantages or disadvantages of a. using these libraries?
iii.	What is the source of the URL that the malware uses for beaconing? What advantages does this source offer?
iv.	Which aspect of the HTTP protocol does the malware leverage to achieve its objectives?
v.	What kind of information is communicated in the malware's initial beacon?
vi.	What are some disadvantages in the design of this malware's communication channels?
vii.	Is the malware's encoding scheme standard?
viii.	How is communication terminated?
ix.	What is the purpose of this malware, and what role might it play in the attacker's arsenal?
	c. This lab builds on Practical 8 a. Imagine that this malware is an attempt by the attacker to improve his techniques. Analyze the malware found in file *Lab14-03.exe*.
i.	What hard-coded elements are used in the initial beacon? What elements, if any, would make a good signature?
ii.	What elements of the initial beacon may not be conducive to a longlasting signature?
iii.	How does the malware obtain commands? What example from the chapter used a similar methodology? What are the advantages of this technique?
iv.	When the malware receives input, what checks are performed on the input to determine whether it is a valid command? How does the attacker hide the list of commands the malware is searching for?

	v.	What type of encoding is used for command arguments? How is it different from Base64, and what advantages or disadvantages does it offer?
	vi.	What commands are available to this malware?
	vii.	What is the purpose of this malware?
	viii.	This chapter introduced the idea of targeting different areas of code with independent signatures (where possible) in order to add resiliency to network indicators. What are some distinct areas of
		a. code or configuration data that can be targeted by network signatures?
	ix.	What set of signatures should be used for this malware?
	d. Analyze the sample found in the file Lab15-01.exe. This is a command-line program that takes an argument and prints "Good Job!" if the argument matches a secret code.	
	i.	What anti-disassembly technique is used in this binary?
	ii.	What rogue opcode is the disassembly tricked into disassembling?
	iii.	How many times is this technique used?
	iv.	What command-line argument will cause the program to print "Good Job!"?
	e. Analyze the malware found in the file Lab15-02.exe. Correct all anti-disassembly countermeasures before analyzing the binary in order to answer the questions.	
	i.	What URL is initially requested by the program?
	ii.	How is the User-Agent generated?
	iii.	What does the program look for in the page it initially requests?
	iv.	What does the program do with the information it extracts from the page?
	f. Analyze the malware found in the file *Lab15-03.exe*. At first glance, this binary appears to be a legitimate tool, but it actually contains more functionality than advertised.	
	i.	How is the malicious code initially called?
	ii.	What does the malicious code do?
	iii.	What URL does the malware use?
	iv.	What filename does the malware use?
9.	**a. Analyze the malware found in *Lab16-01.exe* using a debugger. This is the same malware as *Lab09-01.exe*, with added anti-debugging techniques.**	
	i.	Which anti-debugging techniques does this malware employ?
	ii.	What happens when each anti-debugging technique succeeds?
	iii.	How can you get around these anti-debuggingtechniques?
	iv.	How do you manually change the structures checked during runtime?
	v.	Which OllyDbg plug-in will protect you from the anti-debugging tech- niques used by this malware?
	b. Analyze the malware found in *Lab16-02.exe* using a debugger. The goal of this lab is	

	to figure out the correct password. The malware does not drop a mali- cious payload.
i.	What happens when you run *Lab16-02.exe* from the command line?
ii.	What happens when you run *Lab16-02.exe* and guess the command-line parameter?
iii.	What is the command-line password?
iv.	Load *Lab16-02.exe* into IDA Pro. Where in the mainfunction is strncmp
v.	found?
vi.	What happens when you load this malware into OllyDbg using the default settings?
vii.	What is unique about the PE structure of *Lab16-02.exe*?
viii.	Where is the callback located? (Hint: Use CTRL-E in IDA Pro.)
ix.	Which anti-debugging technique is the program using to terminate immediately in the debugger and how can you avoid this check?
x.	What is the command-line password you see in the debugger after you disable the anti-debugging technique?
xi.	Does the password found in the debugger work on the command line?
c.	**Analyze the malware in *Lab16-03.exe* using a debugger. This malware is similar to *Lab09-02.exe*, with certain modifications, including the introduction of anti-debugging techniques.**
i.	Which strings do you see when using static analysis on the binary?
ii.	What happens when you run this binary?
iii.	How must you rename the sample in order for it to runproperly?
iv.	Which anti-debugging techniques does this malware employ?
v.	For each technique, what does the malware do if it determines it is running in a debugger?
vi.	Why are the anti-debugging techniques successful in this malware?
vii.	What domain name does this malware use?
d.	**Analyze the malware found in *Lab17-01.exe* inside VMware. This is the same malware as *Lab07- 01.exe*, with added anti-VMware techniques.**
i.	What anti-VM techniques does this malware use?
ii.	If you have the commercial version of IDA Pro, run the IDA Python script from Listing 17- 4 in Chapter 17 (provided here as *findAntiVM.py*). What does it find?
iii.	What happens when each anti-VM technique succeeds?
iv.	Which of these anti-VM techniques work against your virtual machine?
v.	Why does each anti-VM technique work or fail?
vi.	How could you disable these anti-VM techniques and get the malware to run?
e.	**Analyze the malware found in the file *Lab17-02.dll* inside VMware. After answering the first question in this lab, try to run the installation exports using *rundll32.exe* and monitor them with a tool like procmon. The following is an example command line for**

	executing the DLL: rundll32.exe Lab17-02.dll,InstallRT (or InstallSA/InstallSB)	
	i.	What are the exports for this DLL?
	ii.	What happens after the attempted installation using *rundll32.exe*?
	iii.	Which files are created and what do they contain?
	iv.	What method of anti-VM is in use?
	v.	How could you force the malware to install during runtime?
	vi.	How could you permanently disable the anti-VMtechnique?
	vii.	How does each installation export function work?
	f. Analyze the malware *Lab17-03.exe* inside VMware.	
	i.	What happens when you run this malware in a virtual machine?
	ii.	How could you get this malware to run and drop its keylogger?
	iii.	Which anti-VM techniques does this malware use?
	iv.	What system changes could you make to permanently avoid the anti-VM techniques used by this malware?
	v.	How could you patch the binary in OllyDbg to force the anti-VM techniques to permanently fail?
10.	**a. Analyze the file *Lab19-01.bin* using *shellcode_launcher.exe***	
	i.	How is the shellcode encoded?
	ii.	Which functions does the shellcode manually import?
	iii.	What network host does the shellcode communicate with?
	iv.	What filesystem residue does the shellcode leave?
	v.	What does the shellcode do?
	b. The file *Lab19-02.exe* contains a piece of shellcode that will be injected into another process and run. Analyze this file.	
	i.	What process is injected with the shellcode?
	ii.	Where is the shellcode located?
	iii.	How is the shellcode encoded?
	iv.	Which functions does the shellcode manually import?
	v.	What network hosts does the shellcode communicate with?
	vi.	What does the shellcode do?
	c. Analyze the file *Lab19-03.pdf*. If you get stuck and can't find the shellcode, just skip that part of the lab and analyze file *Lab19-03_sc.bin* using *shellcode_launcher.exe*.	
	i.	What exploit is used in this PDF?
	ii.	How is the shellcode encoded?

iii.	Which functions does the shellcode manually import?
iv.	What filesystem residue does the shellcode leave?
v.	What does the shellcode do?
d. The purpose of this first lab is to demonstrate the usage of the this pointer. Analyze the malware in *Lab20-01.exe*.	
i.	Does the function at 0x401040 take any parameters?
ii.	Which URL is used in the call to URLDownloadToFile?
iii.	What does this program do?
e. Analyze the malware In Lab20-02.exe.	
i.	What can you learn from the interesting strings in this program?
ii.	What do the imports tell you about this program?
iii.	What is the purpose of the object created at 0x4011D9? Does it have any virtual functions?
iv.	Which functions could possibly be called by the call [edx]instruction at 0x401349?
v.	How could you easily set up the server that this malware expects in order to fully analyze the malware without connecting it to the Internet?
vi.	What is the purpose of this program?
vii.	What is the purpose of implementing a virtual function call in this program?
f. Analyze the malware in Lab20-03.exe.	
i.	What can you learn from the interesting strings in this program?
ii.	What do the imports tell you about this program?
iii.	At 0x4036F0, there is a function call that takes the string Config error, followed a few instructions later by a call to CxxThrowException. Does the function take any parameters other than the string? Does the function return anything? What can you tell about this
a.	function from the context in which it's used?
iv.	What do the six entries in the switch table at 0x4025C8 do?
v.	What is the purpose of this program?
g. Analyze the code in *Lab21-01.exe*	
i.	What happens when you run this program without any parameters?
ii.	Depending on your version of IDA Pro, main may not be recognized automatically. How can you identify the call to the main function?
iii.	What is being stored on the stack in the instructions from 0x0000000140001150 to 0x0000000140001161?
iv.	How can you get this program to run its payload without changing the filename of the executable?
v.	Which two strings are being compared by the call to strncmp at 0x0000000140001205?
vi.	Does the function at 0x00000001400013C8 take any parameters?

	vii.	How many arguments are passed to the call to CreateProcess at 0x0000000140001093? How do you know?
	h. Analyze the malware found in _Lab21-02.exe_ on both x86 and x64 virtual machines.	
	i.	What is interesting about the malware's resource sections?
	ii.	Is this malware compiled for x64 or x86?
	iii.	How does the malware determine the type of environment in which it is running?
	iv.	What does this malware do differently in an x64 environment versus an x86 environment?
	v.	Which files does the malware drop when running on an x86 machine? Where would you find the file orfiles?
	vi.	Which files does the malware drop when running on an x64 machine? Where would you find the file orfiles?
	vii.	What type of process does the malware launch when run on an x64 system?
	viii.	What does the malware do?

Course Outcome	
Learner will be able to	
CO1	Understand various introductory techniques of malware analysis and creating the testing environment
CO2	Perform advanced dynamic analysis and recognize constructs in assembly code.
CO3	Perform Reverse Engineering using OLLYDBG and WINDBG and study the behaviours and functions of malware
CO4	Understand data encoding, various techniques for anti-disassembly and anti-debugging
CO5	Understand various anti virtual machine techniques and perform shellcode analysis of various languages along with x64 architecture.

M. Sc (Information Technology)	Semester – II
Course Name: Cloud Management Practical	**Course Code:** VGVPSTELP202
Periods per week (1 Period is 120 minutes)	2
Credits	2

		Hours	Marks
Evaluation System	**Practical Examination**	3	**100**

Course Objective
To make learners
1. Understand the process of creation and management of the cloud.
2. Get the knowledge of deployment of server manager, management of Configuration Manager etc.
3. Understand the working of an Orchestrator.
4. Get the knowledge of handling of DPM.
5. Understand System Center 2019 and its different components.

List of Practical:	
1.	a. Create and Manage Cloud using SCVMM 2019 b. Deploy a guarded host fabric using Microsoft SCVMM 2019
2.	a. Deploy and manage SDN Infra structure using SCVMM 2019 b. Deploy and Manage Storage Space Direct (S2D) using SCVMM 2019
3.	a. Deploy Service Manager 2019 and install on 4 Computer Scenario b. Setup SQL Server reporting Service using Service Manager 2019

4,	a. User Connectors to import data:
	   i. Import data from Active Directory Domain Services
	   ii. Import data and alerts from Operations Manager
	   iii. Import data from Configuration Manager
	   iv. Import runbooks from Orchestrator
	   v. Import data from VMM
	   vi. Use a CSV file to import data
	b. Automate IT processes with workflows
	   i. Add or remove workflow activities
	   ii. Configure the way activities manage and pass information
	   iii. Deploy a workflow to Service Manager using the Authoring Tool
	   iv. Configure the Activities Toolbox in the Authoring Tool

5.	a. Managing devices with Configuration Manager
	b. Design a hierarchy of sites using Microsoft End Point Configuration manager.
6.	a. Data transfers between sites
	i. Types of data transfer
	ii. File-based replication
	iii. Database replication
	b. Configure sites and hierarchies
	i. Add site system roles
	ii. Install site system roles
	iii. Install cloud-based distribution points
	iv. Configuration options for site system roles
	v. Database replicas for management points
7.	a. Install Orchestrator.
	b. Create and test a monitor runbook
8.	a. Manage Orchestrator Servers – 1
	i. Runbook permissions
	ii. Back up Orchestrator
	iii. Bench mark
	iv. Optimize performance of .Net activities
	v. Configure runbook throttling
	vi. Recover a database
	b. Manage Orchestrator Servers – 2
	i. Recover web components
	ii. Add an integration pack
	iii. View Orchestrator data with PowerPivot
	iv. Change Orchestrator user groups
	v. Common activity properties
	i. vi. Computer groups
9.	Install and Deploy DPM
	i. Install DPM
	ii. Deploy the DPM protection agent
	iii. Deploy protection groups
	iv. Configure firewall settings
10.	Protect Workloads
	i. Back up Hyper-V virtual machines
	ii. Back up SQL Server with DPM
	iii. Back up file data with DPM
	iv. Backup system state and bare metal

	v. Backup and restore VMware servers
	i. vi. Backup and restore VMM servers

Course Outcome	
Learner will be able to	
CO1	Understand the concepts of VMM, SDN, NAS , HyperV etc.
CO2	Understand and use of Service manager with various deployments that can be performed using it.
CO3	Understand and use SCCM and Demonstrate the use of Configuration Manager.
CO4	Use automation with runbooks and demonstrate the use of Windows Orchestrator.
CO5	Use Data Protection Manager.

M. Sc (Information Technology)		Semester – II	
Course Name: Computer Vision Practical		**Course Code:** VGVPSTELP203	
Periods per week (1 Period is 120 minutes)		2	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	**100**

Course Objective
To make learners
1. Understand concepts of geometric transformations, image stitching, camera calibration
2. To introduce the fundamentals of image formation
3. To develop an appreciation for various issues in the design of computer vision and object recognition systems
4. Understanding of the issues involved in trying to define and simulate perception.
5. Familiarize with specific, well known computer vision methods, algorithms and results.

List of Practical:	
1.	Perform Geometric transformations.
2.	Perform Image Stitching.
3.	Perform Camera Calibration.

4,	Perform the following:
	    a. Face detection
	    b. Object detection
	    c. Pedestrian detection
	    d. Face recognition

5.	Construct 3D model from images.
6.	Implement object detection and tracking from video.
7.	Perform Feature extraction using RANSAC.
8.	Perform Colorization.
9.	Perform Text detection and recognition.
10.	Perform Image matting and Composting.

Course Outcome	
Learner will be able to	
CO1	Understand the basics of computer vision
CO2	Understand and analyse various structure form motion and various estimates of Dense Motion
CO3	Apply various motion models to images and understand computation photography techniques
CO4	Apply Epipolar geometry , Rectification and various other 3D correspondence and Stereo reconstruction techniques
CO5	Understand image-based rendering and reconstruction.

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Computer Vision: Algorithms and Applications	Richard Szeliski	Springer	1st Edition	2010

<div align="center">

Evaluation Scheme

</div>

1. Internal Evaluation (40 marks).

      i. **Test: 1 Class test of 15 marks. (Can be taken online)**

Q	Attempt *any three of* the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

      ii.   **15 marks project/presentation/assignment.**

      iii.   **10 marks: Active participation in the class, overall conduct, attendance.**

2. External Examination: (60marks)(2 credit course)

All questions are compulsory		
Q.1.	**(Based on Unit 1) Attempt *any four of* the following:**	**20**
a.		
b.		
..		
f.		
Q.2.	**(Based on Unit 2) Attempt *any four of* the following:**	**20**
Q.3.	**(Based on whole syllabus) Attempt *any four of* the following:**	**20**

External Examination: (60marks)(4 credit course)

All questions are compulsory		
Q.1.	**(Based on Unit 1) Attempt *any three of* the following:**	**15**
a.		
b.		
...		
f.		
Q.2.	**(Based on Unit 2) Attempt *any three of* the following:**	**15**
Q.3.	**(Based on Unit 3) Attempt *any three of* the following:**	**15**
Q.4.	**(Based on Unit 4) Attempt *any three of* the following:**	**15**

3. Practical Exam: 100marks

A Certified copy journal is essential to appear for the practical examination.

1.	Practical Question 1	**40**
2.	Practical Question 2	**40**
3.	Journal	10
4.	Viva Voce	10

OR

1.	Practical Question	**80**
2.	Journal	10
3.	Viva Voce	10