

Semester-I							
S.No.	Course Code	Course Name	Category	Hours per week			Credits
				L	T	P	
1.		<b>Advanced Computer Networks</b>	PC	3	0	0	3
2.		<b>Fundamentals of Data Science</b>	PC	3	0	0	3
3.		Program Elective Course - I a. Computer Vision b. Machine Learning <b>c. Agile Secure Software Engineering</b>	PE	3	0	0	3
4.		Program Elective Course – II a. Distributed Systems b. Information Security <b>c. Data Visualization</b>	PE	3	0	0	3
5.		<b>Advanced Computer Networks Lab</b>	PC	0	0	4	2
6.		<b>Data Science Lab</b>	PC	0	0	4	2
7.		<b>Research Methodology and IPR</b>	MC	2	0	0	2
8.		<b>Audit Course - I</b>	AC	2	0	0	0
<b>Total</b>							<b>18</b>

Semester-II							
S.No.	Course Code	Course Name	Category	Hours per week			Credits
				L	T	P	
1.		Internet of Things	PC	3	0	0	3
2.		Virtual Reality and Augmented Reality	PC	3	0	0	3
3.		Program Elective Course – III a. Cyber Security & Digital Forensics b. Deep Learning c. Service Oriented Architecture	PE	3	0	0	3
4.		Program Elective Course – IV a. Predictive Data Analytics b. Software Defined Networks c. Randomized Approximation Algorithms	PE	3	0	0	3
5.		Internet of Things Lab	PC	0	0	4	2
6.		Virtual Reality and Augmented RealityLab	PC	0	0	4	2
7.		Technical Seminar	PR	0	0	4	2
8.		Audit Course - II	AC	2	0	0	0
Total							18

Semester-III							
S.No.	Course Code	Course Name	Category	Hours per			Credits
				L	T	P	
1.		<b>Program Elective Course – V</b> a. Block chain Technologies b. Reinforcement Learning c. Software Reliability	PE	3	0	0	3
2.		<b>Open Elective</b> a. Software Development and IT Services	OE	3	0	0	3
3.		Dissertation Phase – I	PR	0	0	20	10
4		Co-Curricular Activities	PR				02
<b>Total</b>							<b>18</b>

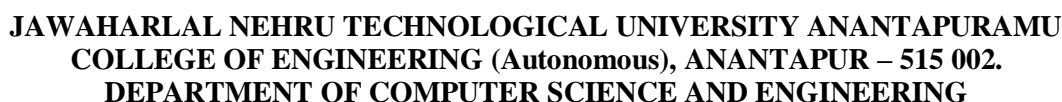


**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPURAMU**  
**COLLEGE OF ENGINEERING (Autonomous), ANANTAPUR – 515 002.**  
**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**M.Tech - Computer Science - I<sup>st</sup> year-I<sup>st</sup> Semester-(R21) Regulations**

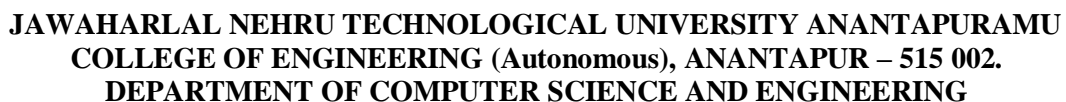
Course Code		Advanced Computer Networks	L	T	P	C
Semester	I		3	0	0	3
Course Objectives:						
<ul style="list-style-type: none"><li>To introduce the wireless and mobile network environment.</li><li>To discuss the working of GSM</li><li>To teach the emerging technologies in the mobile environment.</li><li>To transmit knowledge regarding wireless LANs.</li><li>To elucidate the data dissemination techniques</li><li>To educate the importance of adaptability of the transport layer for the wireless environment.</li></ul>						
Course Outcomes (CO): Student will be able to						
<ul style="list-style-type: none"><li>Describe the mobile network environment</li><li>Justify the need for continuous emergence of technologies</li><li>Experiment with the protocols of GSM environment</li><li>Develop new routing protocols for mobile environment</li><li>Propose new data dissemination techniques</li><li>Recommend modifications to network and transport protocols</li><li></li></ul>						
UNIT – I			Lecture Hrs:			
<b>Introduction:</b> Mobility of Bits and Bytes, Wireless- The Beginning, Mobile Computing, Dialogue Control, Networks, Middleware and Gateways, Application and Services, Developing Mobile Computing Applications, Security in Mobile Computing, Standards-Why are they Necessary?, Standards Bodies, Players in the wireless Space. <b>Mobile Computing Architecture:</b> History of Computers, History of Internet, Internet-The Ubiquitous Network, Architecture for mobile computing, 3-tier architecture, Design considerations for Mobile Computing, Mobile Computing through Internet, Making existing applications mobile-enabled. <b>Mobile Devices and Systems:</b> Mobile Smartphones, Smart mobiles, and Systems, Handheld Packet Computers, Handheld devices, Smart systems, Limitations of Mobile Devices, Automotive Systems						
UNIT – II			Lecture Hrs:			
<b>Emerging Technologies:</b> Introduction, Bluetooth, Radio frequency identification (RFID), Mobile IP, Internet Protocol Version 6(IPV6). Wireless Medium Access Control, CDMA, 3G, Wireless Broadband (WIMAX), 4G and 5G Networks						
UNIT – III			Lecture Hrs:			

<b>Short Message Service(SMS):</b> Mobile Computing over SMS, SMS, Value added Services through SMS. <b>General Packet Radio Service(GPRS):</b> Introduction,GPRS and Packet data Network, GPRS Network Architecture, GPRS Network Operations, Data Services in GPRS, Applications and limitations for GPRS, Billing and charging in GPRS, Enhanced Data rates for GSM Evolution(EDGE). <b>Wireless Application Protocol:</b> Introduction, WAP, MMS.		
<b>UNIT – IV</b>		Lecture Hrs:
<b>Wireless LAN:</b> Introduction, Advantages, IEEE 802.11 Standards, Architecture, Mobility in Wireless LAN, Deploying wireless LAN, Mobile adhoc Networks and Sensor Networks, Wireless LAN Security, Wireless Access in Vehicular environment, Wireless Local loop, Hiper LAN. <b>Mobile Network Layer:</b> Mobile Internet Protocol, Packet delivery and Handover management, Location management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP, VOIP, IPSec.		
<b>UNIT – V</b>		Lecture Hrs:
<b>Mobile Transport Layer:</b> Conventional TCP/IP transport layer protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other methods for Mobile TCP layer Transmission. <b>Data Dissemination and Systems for Broadcasting:</b> Communication Asymmetry, Classification of data delivery mechanism, Data dissemination Broadcast Models, Selective tuning and Indexing Techniques, Digital Audio Broadcasting(DAB), Digital Video Broadcasting		
<b>Textbooks:</b>		
1. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, “Mobile Computing: Technology, Applications and Service Creation”, 2 <sup>nd</sup> edition, McGraw Hill, 2010. 2. RaJ Kamal, “Mobile Computing”, 3 <sup>rd</sup> edition, Oxford University Press. 3.		
<b>Reference Books:</b>		
1. Jochen Schiller, “Mobile Communications”, Addison-Wesley, Second Edition, 2009. 2. UWE Hansmann, Lothar Merk, Martin S. Nocklous, Thomas Stober, “Principles of Mobile Computing,” Second Edition, Springer.		
<b>Online Learning Resources:</b>		



<b>Course Code</b>	<b>I</b>	<b>Fundamentals of Data Science</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Semester</b>	<b>I</b>		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>
<b>Course Objectives:</b>						
<ul style="list-style-type: none"> <li>• Provide you with the knowledge and expertise to become a proficient data scientist.</li> <li>• Demonstrate an understanding of statistics and machine learning concepts that are vital for data science</li> <li>• Produce Python code to statistically analyse a dataset</li> <li>• Critically evaluate data visualizations based on their design and use for communicating stories from data</li> </ul>						
<b>Course Outcomes (CO):</b> Student will be able to						
<ul style="list-style-type: none"> <li>• Understand the key concepts in data science, including their real-world applications and the toolkit used by data scientists</li> <li>• Explain how data is collected, managed and stored for data science</li> <li>• Demonstrate an understanding of statistics and machine learning concepts that are vital for data science</li> <li>• Critically evaluate data visualisations based on their design and use for communicating stories from data</li> </ul>						
<b>UNIT – I</b>		<b>Lecture Hrs:</b>				
Introduction, What Is Statistical Learning?, Why Estimate f?, How Do We Estimate f?, The Trade-Off Between Prediction Accuracy and Model Interpretability, Supervised Versus Unsupervised Learning, Regression Versus Classification Problems, Assessing Model Accuracy, Measuring the Quality of Fit, The Bias-Variance Trade-of, The Classification Setting, Introduction to R, Basic Commands, Graphics, Indexing Data, Loading Data, Additional Graphical and Numerical Summaries.						
<b>UNIT – II</b>		<b>Lecture Hrs:</b>				
Linear Regression, Simple Linear Regression, Multiple Linear Regression, Other Considerations in the Regression Model, Comparison of Linear Regression with K-Nearest Neighbours, Linear Regression.						
<b>UNIT – III</b>		<b>Lecture Hrs:</b>				
Classification, Logistic Regression, Linear Discriminant Analysis, A Comparison of Classification Methods, Logistic Regression, LDA, QDA, and KNN.						
<b>UNIT – IV</b>		<b>Lecture Hrs:</b>				
Programming for basic computational methods such as Eigen values and Eigen vectors, sparse matrices, QR and SVD, Interpolation by divided differences. Data Wrangling: Data Acquisition, Data Formats, Imputation, The split-apply-combine paradigm.						

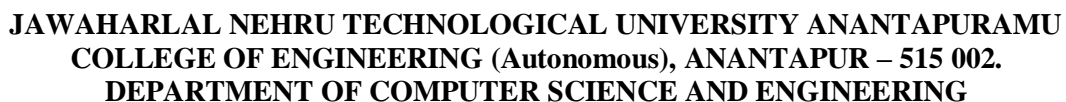
<b>UNIT – V</b>		Lecture Hrs:
Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction.		
<b>Textbooks:</b>		
1. Gareth James Daniela Witten Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, February 11, 2013, web link: <a href="http://www.statlearning.com">www.statlearning.com</a> . 2. Mark Gardener, Beginning R The statistical Programming Language, Wiley, 2015. 3. Han , Kamber, and J Pei, Data Mining Concepts and Techniques, 3rd edition, Morgan Kaufman, 2012.		
<b>Reference Books:</b>		
1. Sinan Ozdemir, Principles of Data Science, Packt Publishing Ltd Dec 2016. 2. Joel Grus, Data Science from Scratch, Oreilly media, 2015.		
<b>Online Learning Resources:</b>		



<b>UNIT - I</b>	<b>Lecture Hrs:</b>
<p><b>Getting started with Security:</b> Security is about Risk, Threat actors and Knowing Your Enemy, Security Values: Protecting our data, systems and people, Common security misconceptions or mistakes.</p> <p><b>Agile Enablers:</b> Build Pipeline, Automated Testing, Continuous Integration, Infrastructure as code, Release Management, Visible Tracking, Centralised Feedback, The Only Good code is deployed code, Operating Safely and at Speed.</p> <p><b>Welcome to the Agile Revolution:</b> Agile: A Potted Landscape, Scrum, the Most Popular of Agile Methodologies ,Extreme Programming, Kanban, Lean ,Agile Methods in General ,What About DevOps? , Agile and Security.</p> <p><b>Working with Your Existing Agile Life Cycle:</b> Traditional Application Security Models, Per-Iteration Rituals, Pre-Iteration Involvement , Post-Iteration Involvement ,Setting Secure Baselines , What About When You Scale? , Building Security Teams That Enable.</p>	
<b>UNIT - II</b>	<b>Lecture Hrs:</b>
<p><b>Security and Requirements:</b> Dealing with Security in Requirements, Agile Requirements: Telling Stories, Tracking and Managing Stories: The Backlog, Dealing with Bugs , Getting Security into Requirements, Security Personas and Anti-Personas , Attacker Stories: Put Your Black Hat On, Attack Trees, Infrastructure and Operations Requirements.</p> <p><b>Agile Vulnerability Management:</b> Vulnerability Scanning and Patching ,Dealing with Critical Vulnerabilities, Securing Your Software Supply Chain, How to Fix Vulnerabilities in an Agile Way ,Security Sprints, Hardening Sprints, and Hack Days, Taking On and Paying Down Security Debt.</p> <p><b>Risk for Agile Teams:</b> Security Says, No, Understanding Risks and Risk Management ,Risks and Threats, Dealing with Risk, Risk Management in Agile and DevOps, Handling Security Risks in Agile and DevOps.</p> <p><b>Threat Assessments and Understanding Attacks:</b> Understanding Threats: Paranoia and Reality, Your System's Attack Surface , Agile Threat Modeling, Common Attack Vectors.</p>	

<b>UNIT - III</b>		Lecture Hrs:
<p><b>Building Secure and Usable Systems:</b> Design to Resist Compromise ,Security Versus Usability , Technical Controls, Security Architecture, Complexity and Security</p> <p><b>Code Review for Security:</b> Why Do We Need to Review Code?, Types of Code Reviews, Peer Code Reviews, When Should You Review Code?, How to Review Code, Who Needs to Review Code?, Automated Code Reviews, Code Review Challenges and Limitations, Adopting Secure Code Reviews, Reviewing Security Features and Controls, Reviewing Code for Insider Threats.</p> <p><b>Agile Security Testing:</b> How Is Testing Done in Agile? , If You Got Bugs, You'll Get Pwned, The Agile Test Pyramid , Unit Testing and TDD, Service-Level Testing and BDD Tool, Acceptance Testing, Functional Security Testing and Scanning, Testing Your Infrastructure, Creating an Automated Build and Test Pipeline, A Place for Manual Testing in Agile, How Do You Make Security Testing Work in Agile and DevOps?</p>		
<b>UNIT - IV</b>		Lecture Hrs:
<p><b>External Reviews, Testing, and Advice:</b> Why Do We Need External Reviews?, Vulnerability Assessment, Penetration Testing, Red Teaming, Bug Bounties, How Bug Bounties ,Configuration Review, Secure Code Audit, Crypto Audit, Choosing an External Firm, Getting Your Money's Worth.</p> <p><b>Operations and OpSec:</b> System Hardening: Setting Up Secure Systems, Network as Code, Monitoring and Intrusion Detection, Catching Mistakes at Runtime , Runtime Defense, Incident Response: Preparing for Breaches, Securing Your Build Pipeline, Shh...Keeping Secrets Secret.</p> <p><b>Compliance:</b> Compliance and Security, Different Regulatory Approaches, Which Approach Is Better? ,Risk Management and Compliance, Traceability of Changes, Data Privacy, How to Meet Compliance and Stay Agile, Building Compliance into Your Culture, Certification and Attestation.</p>		
<b>UNIT – V</b>		Lecture Hrs:
<p><b>Security Culture:</b> The Importance of Security Culture, Building a Security Culture , Principles of Effective Security, Security Outreach.</p> <p><b>What Does Agile Security Mean?:</b> Laura's Story, Jim's Story, Michael's Story, Rich's Story.</p>		
<b>Textbooks:</b>		
1. "Agile Application Security: Enabling Security in a Continuous Delivery Pipeline", By Laura Bell, Michael Brunton-Spall, Rich Smith, Jim Bird, O'Reilly Media, Inc.,2017		
<b>Reference Books:</b>		
<b>Online Learning Resources:</b>		



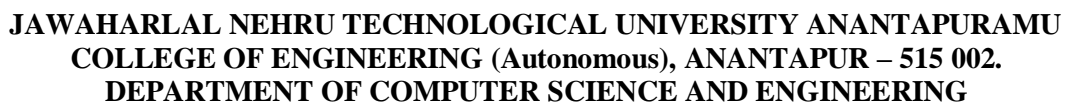


<b>Course Code</b>		<b>Data Visualization</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Semester</b>	<b>I</b>		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>
<b>Course Objectives:</b>						
<ul style="list-style-type: none"> <li>To develop skills to both design and critique visualizations.</li> <li>To introduce visual perception and core skills for visual analysis.</li> <li>To understand visualization for time-series analysis.</li> <li>To understand visualization for ranking analysis.</li> <li>To understand visualization for deviation analysis..</li> </ul>						
<b>Course Outcomes (CO):</b> Student will be able to						
<ul style="list-style-type: none"> <li>Explain principles of visual perception</li> <li>Apply core skills for visual analysis</li> <li>Apply visualization techniques for various data analysis tasks</li> <li>Design information dashboard</li> </ul>						
<b>UNIT - I</b>			<b>Lecture Hrs:</b>			
Information visualization – effective data analysis – traits of meaningful data – visual perception – making abstract data visible – building blocks of information visualization – analytical interaction – analytical navigation – optimal quantitative scales – reference lines and regions – trellises and crosstabs – multiple concurrent views – focus and context – details on demand – over-plotting reduction – analytical patterns – pattern examples.						
<b>UNIT - II</b>			<b>Lecture Hrs:</b>			
Distribution analysis – describing distributions – distribution patterns – distribution displays – distribution analysis best practices – correlation analysis – describing correlations – correlation patterns – correlation displays – correlation analysis techniques and best practices – multivariate analysis – multivariate patterns – multivariate displays – multivariate analysis techniques and best practices.						
<b>UNIT - III</b>			<b>Lecture Hrs:</b>			
Information dashboard – Introduction– dashboard design issues and assessment of needs – Considerations for designing dashboard-visual perception – Achieving eloquence.						
<b>UNIT - IV</b>			<b>Lecture Hrs:</b>			
Advantages of Graphics _Library of Graphs – Designing Bullet Graphs – Designing Sparklines – Dashboard Display Media –Critical Design Practices – Putting it all together- Unveiling the dashboard.						
<b>UNIT - V</b>			<b>Lecture Hrs:</b>			
Plotting Geospatial Data: Introduction to Geoplotlib, Design Principles of Geoplotlib, Geospatial Visualizations, Plotting Geospatial Data on a Map Web-Based Visualizations: Concepts of Bokeh, Interfaces-Plotting and Model Interfaces, Output, Bokeh Server, Presentation, Integrating – HTML Document and Bokeh Applications						
<b>Textbooks:</b>						
<ol style="list-style-type: none"> <li>Ben Fry, "Visualizing data: Exploring and explaining data with the processing environment", O'Reilly, 2008.</li> <li>Mario Dobler, Tim Grobmann, “Data Visualization with Python”, O’Reilly, First Edition, 2019</li> </ol>						

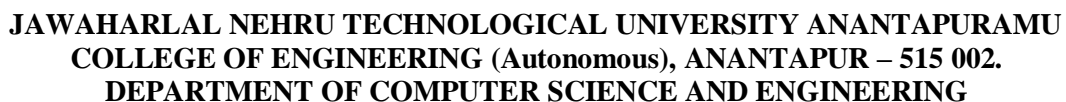
3.Scott Murray, “Interactive data visualization for the web”, O’Reilly Media, Inc., 2013

**Reference Books:**

1. Stephen Few, "Information dashboard design: Displaying data for at-a-glance monitoring", second edition, Analytics Press, 2013.
2. Greg Conti, “Security Data Visualization: Graphical Techniques for Network Analysis”, No Starch Press Inc, 2007.



<b>Course Code</b>		<b>Advanced Computer Networks Lab</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Semester</b>	<b>I</b>		<b>0</b>	<b>0</b>	<b>4</b>	<b>2</b>
<b>Course Objectives:</b>						
<ul style="list-style-type: none"> <li>Aims to provide advanced background on relevant computer networking topics to have a comprehensive and deep knowledge in computer networks</li> </ul>						
<b>Course Outcomes (CO):</b>						
<ul style="list-style-type: none"> <li>Develop programs for client-server application</li> <li>Perform packet sniffing and analyze packets in network traffic.</li> <li>Implement error detecting and correcting codes</li> <li>Implement network security algorithms</li> <li></li> </ul>						
<b>List of Experiments:</b>						
<ol style="list-style-type: none"> <li>Implementation of client server programs for different network applications</li> <li>Study and analysis of the network using Wireshark network protocol analyser</li> <li>Implementation of topology generation for network simulation</li> <li>Implementation of queuing management</li> <li>Implementation of MAC-layer protocols</li> <li>Implementation of routing protocols</li> <li>Implementation of transport-layer protocols</li> <li>Implementation of network security mechanisms</li> </ol>						
<b>References:</b>						
Online learning resources/Virtual labs						



<b>Course Code</b>		<b>Data Science Lab</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Semester</b>	<b>I</b>		<b>0</b>	<b>0</b>	<b>4</b>	<b>2</b>

- To make students understand learn about a Big Data – R Programming, way of solving problems.
- To teach students to write programs in Scala to solve problems.

- Develop new algorithms and software tools for data management and mining, and to use them for social good.
- Applying machine learning, data mining, and network analysis to real-world problems in society and industry.
- Exploring the creation of novel statistical and computational methods for scalable data mining, machine learning, optimization as well as statistical modelling with complex data sets

1. Write an R program to evaluate the following expression  $ax+b/ax-b$ .
2. Write an R program to read input from keyboard (hint: `readLine()`).
3. Write an R program to find the sum of n natural numbers:  $1+2+3+4+\dots+n$
4. Write an R program to read n numbers. (i) Sum of all even numbers (ii) Total number of even numbers.
5. Write an R program to read n numbers. (i) Total number of odd numbers (ii) Sum of all odd numbers.
6. Write an R program to obtain (i)sum of two matrices A and B (ii) subtraction of two matrices A and B (iii) Product of two matrices.
7. Write an R program for “declaring and defining functions “
8. Write an R program that uses functions to add n numbers reading from keyboard .
9. Write an R program uses functions to swap two integers.
10. Write an R program that use both recursive and non-recursive functions for implementing the Factorial of a given number, n.
11. Write an R program to reverse the digits of the given number {example 1234 to be written as 4321 }

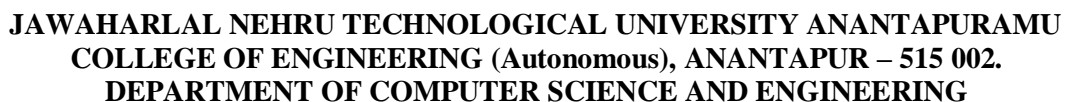
12. Write an R program to implement (i) Linear search (ii) Binary Search.
13. Write an R program to implement (i) Bubble sort (ii) selection sort.
14. Write a R program to implement the data structures (i) Vectors (ii) Array (iii) Matrix (iv) Data Frame (v) Factors
15. Write a R program to implement scan(), merge(), read.csv() and read.table() commands.
16. Write an R program to implement “Executing Scripts” written on the note pad, by calling to the R console.
17. Write a R program, Reading data from files and working with datasets (i) Reading data from csv files, inspection of data. (ii) Reading data from Excel files.
18. Write a R program to implement Graphs (i) Basic high-level plots (ii) Modifications of scatter plots (iii) Modifications of histograms, parallel box plots.

**References:**

1. Big data – Black Book: 2015 edition: dreamtech press. Pg. (490- 642).
2. Introducing to programming and problem solving by scala, mark c. lewis, lisa l.lacher. CRC press, second edition.
- 3.

**Online Learning Resources:**

1. <https://www.tutorialspoint.com/scala/>
2. <https://www.tutorialspoint.com/r/>



Course Code			L	T	P	C
Semester	I	Research Methodology & IPR	2	0	0	2
<b>Course Objectives:</b>						
<ul style="list-style-type: none"> <li>To understand the research problem</li> <li>To know the literature studies, plagiarism and ethics</li> <li>To get the knowledge about technical writing</li> <li>To analyze the nature of intellectual property rights and new developments</li> <li>To know the patent right</li> </ul>						
<b>Course Outcomes (CO):</b> Student will be able to						
<ul style="list-style-type: none"> <li>Understand research problem formulation.</li> <li>Analyze research related information and Follow research ethics</li> <li>Understand that today's world is controlled by Computer, Information Technology, but tomorrow world will be ruled by ideas, concept, and creativity.</li> <li>Understanding that when IPR would take such important place in growth of individuals &amp; nation, it is needless to emphasise the need of information about Intellectual Property Right to be promoted among students in general &amp; engineering in particular.</li> <li>Understand that IPR protection provides an incentive to inventors for further research work and investment in R &amp; D, which leads to creation of new and better products, and in turn brings about, economic growth and social benefits</li> </ul>						
<b>UNIT - I</b>			Lecture Hrs:			
Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations						
<b>UNIT - II</b>			Lecture Hrs:			
Effective literature studies approaches, analysis, Plagiarism, Research ethics						
<b>UNIT - III</b>			Lecture Hrs:			
Effective technical writing, how to write report, Paper Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee						
<b>UNIT - IV</b>			Lecture Hrs:			
Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of Patenting and Development: technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT.						
<b>UNIT - V</b>			Lecture Hrs:			
Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications. New Developments in IPR: Administration of Patent System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.						

<b>Textbooks:</b>
<ol style="list-style-type: none"> <li>1. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science &amp; engineering students"</li> <li>2. Wayne Goddard and Stuart Melville, "Research Methodology: An Introduction"</li> </ol>
<b>Reference Books:</b>
<ol style="list-style-type: none"> <li>1. Ranjit Kumar, 2nd Edition, "Research Methodology: A Step by Step Guide for beginners"</li> <li>2. Halbert, "Resisting Intellectual Property", Taylor &amp; Francis Ltd ,2007.</li> <li>3. Mayall, "Industrial Design", McGraw Hill, 1992.</li> <li>4. Niebel, "Product Design", McGraw Hill, 1974.</li> <li>5. Asimov, "Introduction to Design", Prentice Hall, 1962.</li> <li>6. Robert P. Merges, Peter S. Menell, Mark A. Lemley, "Intellectual Property in New Technological Age", 2016.</li> <li>7. T. Ramappa, "Intellectual Property Rights Under WTO", S. Chand, 2008</li> </ol>
<b>Online Learning Resources:</b>

Semester-II							
S.No.	Course Code	Course Name	Category	Hours per week			Credits
				L	T	P	
1.		Internet of Things	PC	3	0	0	3
2.		Virtual Reality and Augmented Reality	PC	3	0	0	3
3.		Program Elective Course – III d. Cyber Security & Digital Forensics e. Deep Learning f. Service Oriented Architecture	PE	3	0	0	3
4.		Program Elective Course – IV d. Predictive Data Analytics e. Software Defined Networks f. Randomized Approximation Algorithms	PE	3	0	0	3
5.		Internet of Things Lab	PC	0	0	4	2
6.		Virtual Reality and Augmented RealityLab	PC	0	0	4	2
7.		Technical Seminar	PR	0	0	4	2
8.		Audit Course - II	AC	2	0	0	0
Total							18



**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**  
**I M.TECH–I I SEMESTER**  
**Internet of Things** **L-T-P-C: 3-0-0-3**

**Course Objectives:**

- To provide students with good depth of knowledge of Designing Industrial IOT Systems for various applications.
- Knowledge for the design and analysis of Industry 4.0 Systems .

**Course Outcomes:**

CO1: Knowledge of theory and practice related to Industrial IoT Systems.

CO2: Ability to identify, formulate and solve engineering problems by using Industrial IoT.

CO3: Ability to implement real field problem by gained knowledge of Industrial applications with IoT capability.

**UNIT I**

IoT Data Management and Compute Stack , Fog Computing, Edge Computing ,The Hierarchy of Edge, Fog and Cloud , Smart Objects: The “Things” in IoT Sensors, Actuators, and Smart Objects Sensors.

**UNIT II**

Actuators Micro-Electro-Mechanical Systems (MEMS) Smart Objects Smart Objects: A Definition Trends in Smart Objects ,Sensor Networks, Wireless Sensor Networks (WSNs) ,Communication Protocols for Wireless Sensor Networks.

**UNIT III**

Connecting Smart Objects ,Communications Criteria, Range Frequency Bands, Power Consumption Topology ,Constrained Devices, Constrained-Node Networks, Data Rate and Throughput Latency and Determinism Overhead and Payload IoT Access Technologies ,IEEE 802.15.4 ,Standardization and Alliances ,Physical Layer, MAC Layer, Topology Security ,Competitive Technologies .

**UNIT IV**

Smart and Connected Cities, An IoT Strategy for Smarter Cities, Vertical IoT Needs for Smarter Cities, Global vs. Siloed Strategies ,Smart City IoT Architecture ,Street Layer City Layer, Data Center Layer, Services Layer, On-Premises vs. Cloud Smart City Security Architecture , Smart City Use-Case Examples, Connected Street Lighting , Connected Street Lighting Solution Street Lighting .

**UNIT V**

Architecture Smart Parking, Smart Parking Use Cases ,Smart Parking Architecture , Smart Traffic Control, Smart Traffic Control Architecture ,Smart Traffic Applications, Connected Environment, The Need for a Connected Environment, Connected Environment Architecture.

**Text Books**

1. IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, Cisco Press, 2017.
2. Industry 4.0: The Industrial Internet of Things Alasdair Gilchrist Publications: A press.

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**  
**I M.TECH-I SEMESTER**

**Virtual Reality and Augmented Reality**

**L-T-P-C: 3-0-0-3**

**Course Objectives:**

- To Teach about human interaction with computers
- To Demonstrate Virtual reality
- To introduce to the current state of VR Hardware and Software.
- To explain how to apply VR/MR/AR for various applications.

**Course Outcomes:**

CO1: Study the fundamentals of VR, AR and MR

CO2: Select appropriate software and hardware for developing VR Applications

CO3: Design VR Applications

CO4: Create game objects using unity

**UNIT 1: Introduction to Virtual Reality**

What is Virtual Reality, Modern VR experiences, History Repeats.

**Unity:** Virtually Everything for you, what is virtual reality to you, types of head-mounted displays: Desktop VR, Mobile VR, The difference between virtual reality and augmented reality, Applications vs Games, Types of VR experiences, and Technical skills that are important to VR.

**UNIT 2: Bird's-Eye View**

Hardware, Software, Human Physiology and Perception.

**Unity:** Objects and Scale: Getting started with unity, creating a simple Diorama, Measurement tools, First Person Character: Understanding the Unity characters, Unity standard assets.

**UNIT 3: The Geometry of Virtual Worlds & Light and Optics:**

Geometric Models, Changing Position and Orientation, Axis-Angle Representations of Rotation, Viewing Transformations, Chaining the Transformations

**Light and Optics: Basic behavior of light, lenses, Optical Aberrations,** Human Eye, Cameras, and Displays.

**UNIT 4: The Physiology of Human Vision**

From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR.

**UNIT 5: Motion in Real and Virtual Worlds** The Vestibular System, Physics in the Virtual World.

**Audio:** The Physics of Sound, the Physiology of Human Hearing, Auditory Perception

**Text Books:**

1. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016
2. Unity Virtual reality Projects, Jonathan Linowes, PACKT Publishing.

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH–I SEMESTER**  
**(Program Elective Course III)**

**Service Oriented Architecture**

**L-T-P-C: 3-0-0-3**

**Course Objectives:**

- Understand SOA and evolution of SOA.
- Understand web services and primitive, contemporary SOA.
- Understand various service layers.
- Understand service-oriented analysis and design based on guidelines.

**Course Outcomes:**

**CO1:** understand basic principles of service oriented architecture

**CO2:** gain knowledge on web service specifications and standards

**CO3:** build service oriented web applications

**CO4:** apply service layers in developing web services

**UNIT I**

**Introducing SOA:** Fundamental SOA, Common Characteristics of Contemporary SOA, Common Tangible Benefits of SOA, Common Pitfalls of Adopting SOA. The Evolution of SOA: An SOA Timeline, The Continuing Evolution of SOA, The Roots of SOA.

**UNIT II**

**Web Services and Primitive SOA:** The Web Services Frame Work, Services, Service Descriptions, Messaging. Web Services and Contemporary SOA (Part I-Activity management and Composition): Message Exchange Patterns, Service Activity, Coordination, Atomic Transactions, Orchestration, Choreography. Web Services and Contemporary SOA (Part-II-Advanced Messaging, Metadata and Security): Addressing, Reliable Messaging, Correlation, Policies, Metadata exchange, Security.

**UNIT III**

**Principles of Service-Orientation:** Service–Orientation and the Enterprise, Anatomy of SOA, Common Principles of Service–Orientation, Interrelation between Principles of ServiceOrientation, Service Orientation and Object Orientation, Native Web Services Support for Principles of Service-Oriented. Service Layers: Service-Oriented and Contemporary SOA, Service Layer abstraction, Application Service Layer, Business Service Layer, Orchestration Service Layer, Agnostic Services, Service Layer Configuration Scenarios.

**UNIT IV**

**SOA Delivery Strategies:** SOA Delivery Lifecycle Phases, The Top-Down Strategy, The Bottom-up Strategy, The Agile Strategy. Service Oriented Analysis (Part I-Introduction): Introduction to Service Oriented Analysis, Benefits of a Business Centric SOA, Deriving Business Services.

**Service Oriented Analysis (Part-II-Service Modelling):** Service Modeling, Service Modelling Guidelines, Classifying Service Model Logic, Contrasting Service Modeling Approaches.

Service Oriented Design (Part I-Introduction): Introduction to Service-Oriented Design, WSDL Related XML Schema Language Basics, WSDL Language Basics, Service Interface Design Tools. Service Oriented Design (Part II-SOA Composition Guidelines): SOA Composing Steps, Considerations for Choosing Service Layers, Considerations for Positioning Core SOA Standards, Considerations for Choosing SOA Extensions.

## **UNIT V**

**Service Oriented Design (Part III- Service Design):** Service Design Overview, Entity-Centric Business Service Design, Application Service Design, Task-Centric Business Service Design, Service Design Guidelines.

**Service Oriented Design (Part IV-Business Process Design):** WS-BPEL Language Basics, WS- Coordination Overview, Service Oriented Business Process Design.

### **Text Books:**

1. Service-Oriented Architecture-Concepts, Technology, and Design, Thomas Erl, Pearson Education.
2. Understanding SOA with Web Services, Eric Newcomer, Greg Lomow, Pearson Education.

### **Reference Books:**

1. The Definitive guide to SOA, Jeff Davies & others, Apress, Dreamtech.
2. Java SOA Cook book, E.Hewitt, SPD.
3. SOA in Practice, N.M.Josuttis, SPD.
4. Applied SOA, M.Rosen and others, Wiley India pvt. Ltd.
5. Java Web Services Architecture, J.Mc Govern, and others, Morgan Kaufmann Publishers, Elsevier.
6. SOA for Enterprise Applications, Shankar.K, Wiley India Edition.
7. SOA-Based Enterprise Integration, W.Roshen, TMH.
8. SOA Security, K.Rama Rao, C.Prasad, dreamtech press

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH-I SEMESTER**  
**(Program Elective Course IV)**

**Software Defined Networks**

**L-T-P-C: 3-0-0-3**

**Course Objectives:** This course introduces about software defined networking, an emerging paradigm in computer networking that allows a logically centralized software program to control the behavior of an entire network.

**Course outcomes:**

CO1: Differentiate between traditional networks and software defined networks and understand the key benefits and use cases of SDN.

CO2: Interpret the SDN data plane devices and Open-Flow Protocols

CO3: Implement the operation of SDN control plane with different controllers

CO4: Apply techniques that enable applications to control the underlying network using SDN

CO5: Evaluate Network Functions Virtualization components and their roles in SDN

**UNIT 1:**

**Introduction:** basic packet switching terminology, Historical background, the modern data centre, traditional switch architecture, Autonomous and Dynamic forwarding tables, Can we increase the packet forwarding IQ, Open source and technological shifts.

**Why SDN:** Evolution of switches and control panels, Cost, SDN Implications for research and innovation, Data centre innovation, Data centre needs.

**UNIT 2:**

**Genesis of SDN:** The evolution of networking technology, Forerunners of SDN, Legacy mechanisms evolve toward SDN, Software defined networking is born, Sustaining SDN interoperability, Open source contributions, Network virtualization, May I please call my network SDN.

**How SDN Works:** Fundamentals characteristics of SDN, SDN operation, SDN devices, SDN controller, SDN applications, Alternate SDN methods.

**UNIT 3:**

**The OpenFlow Specification:** Chapter-specific terminology, OpenFlow overview, OpenFlow 1.0 and OpenFlow basics, OpenFlow 1.1 additions, OpenFlow 1.2 additions, OpenFlow 1.3 additions, OpenFlow 1.4 additions, OpenFlow 1.5 additions, Improving OpenFlow interoperability, Optical transport protocol extensions, OpenFlow limitations.

**UNIT 4:**

**SDN in the data centre:** Data centre definition, Data centre demands, Tunnelling technologies for the data centre, Path technologies in the data centre, Ethernet fabrics in the data centre, SDN use cases in the data centre, Comparison of open SDN, Overlays and APIs, Real-world data centre implementations.

**Network Function Virtualization:** Definition of NFV, what can we virtualize, Standards, OPNFV, Leading NFV vendors, SDN vs NFV, In-Line Network functions.

#### **UNIT 5:**

**SDN Applications:** Terminology, before you begin, Application Types, A brief history of SDN controllers, Using floodlight for training purposes, A simple reactive java application, Controller considerations, Network device considerations, Creating network virtualization tunnels, Offloading flows in the data centre, Access control for the campus, traffic Engineering for service providers.

**SDN Futures:** Current state of affairs, SD-WAN, Potential novel applications of Open SDN.

#### **Textbooks:**

1. Paul Goransson Chuck Black timothy Culver: Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, Second Edition.

#### **References:**

1. Ken Gray Thomas Nadeau: network Function Virtualization, Morgan Kaufmann, 2016.

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH-I SEMESTER**

**Internet of Things Lab**

**L-T-P-C: 0-0-4-2**

**Course Objectives:**

The main objective IOT applications are to know the different real time sensors used to measure the different electrical parameters and to control the different devices from anywhere through IOT.

**Course Outcomes:**

CO1: The students will be thorough about the technology behind the IoT and associated technologies

CO2: The students will be able to use the IoT technologies in practical domains of society

CO3: The students will be able to gain knowledge about the state of the art methodologies in IoT application domains.

1. Setting up of Raspberry Pi and connect to a network.
2. Familiarization with GPIO pins and control hardware through GPIO pins.
3. Speed Control of motors using PWM with python programming.
4. Use sensors to measure temperature, humidity, light and distance.
5. Web based hardware control.
6. Connect IOT devices through cloud using IoT protocol such as MQTT.
7. Controlling IoT devices using Arduino.
8. Create Wireless network of sensors using Zigbee.
9. Experiment on connectivity of Raspberry Pi with existing system components.
10. Exercise on working principle of Raspberry Pi.

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**  
**I M.TECH-I SEMESTER**

**Virtual reality & Augmented Reality Lab    L-T-P-C: 0-0-4-2**

**Course Objective:**

The objective of this course is to explore the concepts of Virtual reality and develop 3D virtual environment.

**Course Outcomes:**

CO1: Create and deploy a VR application.

CO2: understand the physical principles of VR

CO3: Create a comfortable, high-performance VR application using Unity

CO4: Identify, examine and develop software that reflects fundamental techniques for the design and deployment of VR experiences.

**List of Programs:**

1. Installation of Unity and Visual Studio, setting up Unity for VR development, understanding documentation of the same.
2. Demonstration of the working of HDMs
3. Develop a scene in Unity that includes:
  - i. a cube, plane and sphere, apply transformations on the 3 game objects.
  - ii. add a video and audio source
4. Develop a scene in Unity that includes a cube, plane and sphere. Create a new material and texture separately for three Game objects. Change the colour, material and texture of each Game object separately in the scene. Write a C# program in visual studio to change the colour and material/texture of the game objects dynamically on button click.
5. Develop a scene in Unity that includes a sphere and plane . Apply Rigid body component, material and Box collider to the game Objects. Write a C# program to grab and throw the sphere using vr controller.
6. Develop a simple UI(User interface ) menu with images, canvas, sprites and button. Write a C# program to interact with UI menu through VR trigger button such that on each successful trigger interaction display a score on scene.
7. Create a Simple Mini Project



Semester-III							
S.No.	Course Code	Course Name	Category	Hours per			Credits
				L	T	P	
1.		<b>Program Elective Course – V</b> d. Block chain Technologies e. Reinforcement Learning f. Software Reliability	PE	3	0	0	3
2.		<b>Open Elective</b> b. Software Development and IT Services	OE	3	0	0	3
3.		Dissertation Phase – I	PR	0	0	20	10
4		Co-Curricular Activities	PR				02
<b>Total</b>							<b>18</b>

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH-I SEMESTER**  
**(Program Elective Course V)**

**Block Chain technologies**

**L-T-P-C: 3-0-0-3**

**Course Objectives:** This course is intended to study the basics of Block chain technology. During this course learner will explore various aspects of Block chain technology like application in various domains. By implementing learner will have idea about private and public Block chain, and smart contract.

**Course outcomes:**

CO1: Understand and explore the working of Blockchain technology (Understanding)

CO2: Analyze the working of Smart Contracts (Analyze)

CO3: Understand and analyze the working of Hyperledger (Analyze)

CO4: Apply the learning of solidity and de-centralized apps on Ethereum (Apply).

**UNIT 1:**

**Block chain basics:** From Bit-coin to block chain, what is a block chain? Block chain programming, Motivating scenarios.

**Smart contracts:** The concept of a smart contract, Design of a smart contract, Development of a smart contract code, Deploying and testing the smart contract, What makes a block chain contract smart? Decentralized airline system use case, Airlines smart contract, smart contract design considerations.

**UNIT 2:**

**Techniques for trust and integrity:** Essentials of trust and integrity, Digital democracy problem, Testing, Using modifiers, require(), and revert(), Assert() declarations.

**From smart contracts to Dapps:** Dapp development using Truffle, Install ganache test Chain, Develop the smart contract, develop and configure the web application

**UNIT 3:**

**Security and privacy:** Cryptography basics, The relevance of public-key cryptography to blockchain, Hashing basics, Application of hashing.

**On-Chain and off-chain data:** On-chain data, Blind auction use case, Off-chain data: External data sources, ASK airlines system.

**Web3 and a channel Dapp:** Web3 API, The channel concept, Micropayment channel, Micropayment channel use case.

#### **UNIT 4:**

**Going public with Infura:** Nodes and networks, Infura blockchain infrastructure, Going public with Infura, End-to-end process for public development, Deploying BlindAuction-Dapp on Infura, Deploying MPC-Dapp on Infura.

**Tokenization of assets:** Ethereum standards, RES\$: Non-fungible real estate token.

**Testing smart contracts:** Importance of testing smart contracts, Testing counter smart contract, Testing ballot smart contract, TRecap writing of test script, The blind auction test script.

#### **UNIT 5:**

**A roadmap to Dapp development:** Motivating scenario: Educational credentialing, The roadmap, problem description, Analysis and design, Developing the smart contract, local deployment, Automated testing using truffle, Developing the web application, Testing the DCC-Dapp.

**Blockchain:** The road ahead: Decentralized identity, self-managed identity, consensus and integrity, scalability, scalability solutions, privacy, public, private and permissioned networks, confidentiality, security, securing it with cryptocurrency, Accessing off-chain data(Oracles), from foundations to practical systems, Looking ahead.

#### **Textbooks:**

2. Bina Ramamurthy: Blockchain in Action: Manning publications (2020).

#### **References:**

2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press (July 19, 2016).
3. DR. Gavin Wood, "ETHEREUM: A Secure Decentralized Transaction Ledger,"Yellow paper.2014.

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH–I SEMESTER (R20)**

(Program Elective Course V)

**Reinforcement Learning**

**L-T-P-C: 3-0-0-3**

**Course Objectives:**

- Reinforcement Learning is a subfield of Machine Learning, but is also a general-purpose formalism for automated decision-making and AI. This course introduces you to statistical learning techniques where an agent explicitly takes actions and interacts with the world.

**Course Outcomes:**

CO1: Formulate Reinforcement Learning problems

CO2: Apply various Tabular Solution Methods to Markov Reward Process Problems

CO3: Apply various Iterative Solution methods to Markov Decision Process Problems

CO4: Comprehend Function approximation methods

**UNIT 1:**

Introduction: Introduction to Reinforcement Learning (RL) – Difference between RL and Supervised Learning, RL and Unsupervised Learning. Elements of RL, Markov property, Markov chains, Markov reward process (MRP).

**UNIT 2:**

Evaluative Feedback - Multi-Arm Bandit Problem: An n-Armed Bandit Problem, Exploration vs Exploitation principles, Action value methods, Incremental Implementation, tracking a non-stationary problem, optimistic initial values, upper-confidence-bound action selection, Gradient Bandits. Introduction to and proof of Bellman equations for MRPs

**UNIT 3:**

Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations. Dynamic Programming (DP): Overview of dynamic programming for MDP, principle of optimality, Policy Evaluation, Policy Improvement, policy iteration, value iteration, asynchronous DP , Generalized Policy Iteration.

**UNIT 4:**

Monte Carlo Methods for Prediction and Control: Overview of Monte Carlo methods for model free RL, Monte Carlo Prediction, Monte Carlo estimation of action values, Monte Carlo Control, On policy and off policy learning, Importance sampling. Temporal Difference Methods: TD Prediction, Optimality of TD(0), TD Control methods - SARSA, Q-Learning and their variants.

**UNIT 5:**

Eligibility traces: n-Step TD Prediction, Forward and Backward view of  $TD(\lambda)$ , Equivalence of forward and backward view,  $Sarsa(\lambda)$ , Watkins's  $Q(\lambda)$ , Off policy eligibility traces using importance of sampling. Function Approximation Methods: Value prediction with function approximation, gradient descent methods, Linear methods, control with function approximation.

**Textbooks:**

1. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction", 2nd Edition, The MIT Press.
2. Csaba Szepesvari – Algorithms for Reinforcement Learning – Morgan & Claypool, 2010.

**References:**

1. Reinforcement Learning By Richard S. (University Of Alberta) Sutton, Andrew G. (Co-Director Autonomous Learning Laboratory) Barto

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**  
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH-I SEMESTER**  
**(Program Elective Course V)**

**Software Reliability**

**L-T-P-C: 3-0-0-3**

**Course Objectives:**

- To understand concepts of quality & reliability
- To evaluate the overall reliability of a system from component reliability

**Course Outcomes:**

CO1: Model repairable and non-repairable systems and calculate failure rate, repair rate, reliability and availability

CO2: Use various probability density distributions significant to reliability calculations

CO3: Fit a given failure data set of a product into a Weibull distribution and estimate the reliability parameters

**UNIT 1:**

**Concepts of Product Quality:** Quality Function Deployment / House of Quality, Six Sigma

**UNIT 2:**

**Concepts of Reliability:** Basic concepts of repairable and non-repairable systems, Reliability, Availability and Maintainability

**UNIT 3:**

**Failure data analysis:** Fitting discrete and continuous distributions to failure data sets, Weibull analysis, estimation of important reliability parameters

**UNIT 4:**

**Calculation of System Reliability from Component reliabilities:** Markov modelling of repairable and non-repairable systems, Reliability Logic Diagrams, Fault-tree analysis

**UNIT 5:**

**Preventive and Predictive maintenance:** Failure Modes and Effects Analysis.

**Textbooks:**

1. Louis Cohen, Joseph P. Ficalora, Quality Function Deployment and Six Sigma: A QFD Handbook, Prentice Hall, Second Edition, 2009, ISBN: 9780137035441
2. VNA Naikan, Reliability Engineering and Life Testing, PHI Learning, 2010, ISBN: 978-8120335936
3. Singiresu S Rao, Reliability Engineering, Pearson Education, 2014, ISBN: 978-0136015727

**References:**

1. Patrick O Connor, Practical Reliability Engineering, John Wiley, Student ed., 2009, ISBN: 9780470979815
2. B.L. Hansen & P.M. Ghare, Quality Control and Applications, Prentice-Hall, 1997, ISBN: 9780137452255

**JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS):: ANANTHAPURAMU**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**I M.TECH-I SEMESTER**

**(Open Elective)**

**Software Development and IT Services**

**L-T-P-C: 3-0-0-3**

**Course objectives:**

- Take user stories and translate them into functioning web applications using HTML, CSS, and JavaScript
- Evaluate alternative approaches to software implementations
- Work through coding issues with analytical debugging techniques

**Course Outcomes:**

**Unit – I :**

**The Big Picture:** A Snapshot of Devops Culture, The Evolution of Culture, The Value of the Story, Illustrating Devops with Stories, What is Devops? The Devops equation, A History of Devops, Developer as Operator , The Advent of Software Engineering , The Advent of Proprietary Software and Standardization , The Age of the Network, The Beginnings of a Global Community ,The Age of Applications and the Web , The Growth of Software Development Methodologies , Open Source Software, Proprietary Services , Agile Infrastructure , The Beginning of devopsdays , The Current State of Devops .

**Foundational Terminology and Concepts:** Software Development Methodologies, Operations Methodologies, Systems Methodologies, Development, Release, and Deployment Concepts, Infrastructure Concepts, Cultural Concepts

**Devops Misconceptions and Anti-Patterns:** Common Devops Misconceptions, Devops Anti-Patterns, The Four Pillars of Effective Devops

**Unit 2:**

**Collaboration:** Individuals Working Together, Defining Collaboration, Individual Differences and Backgrounds, Opportunities for Competitive Advantage, Mentorship, Introducing Mindsets, Mindsets and Learning Organizations, the Role of Feedback, Reviews and Rankings, Communication and Conflict Resolution Styles, Communication Context and Power Differentials, Empathy and Trust, Humane Staffing and Resources, Effective Collaboration with Sparkle Corp.

**Collaboration: Misconceptions and Troubleshooting:** Collaboration Misconceptions, Collaboration Troubleshooting.

### **Unit – III:**

**Affinity:** From Individuals to Teams, What Makes a Team, Teams and Organizational Structure, Finding Common Ground Between Teams, Improving Team Communication, Case Study: United States Patent and Trademark Office, Benefits of Improved Affinity, Requirements for Affinity, Measuring Affinity

**Misconceptions and Troubleshooting:** Affinity Misconceptions, Affinity Troubleshooting.

### **Unit – IV:**

Overview of Software, Automation, Monitoring, Metrics, Logging, Alerting, Events, Evolution of the Ecosystem.

**Tools:** Accelerators of Culture, What Are Tools? Irrelevance of Tools, Selection of Tools, Auditing Your Tool Ecosystem, Case Studies, Examining Etsy, Motivations and Decision-Making Challenges.

### **UNIT-V:**

**Scaling:** Inflection Points, Understanding Scaling, Organizational Structure, Team Flexibility, Organizational Lifecycle, Complexity and Change, Scaling for Teams.

**Case Studies:** Growing and Scaling Teams, Job Postings and Recruitment Issues, Developing Individuals and Teams, Team Scaling and Growth Strategies, Managing Conflict, Scaling for Organizations.

**Misconceptions and Troubleshooting:** Scaling Misconceptions, Scaling Troubleshooting.

### **TEXT BOOKS:**

1. Effective DevOps Building a Culture of Collaboration, Affinity, and Tooling at Scale, Jennifer Davis and Ryn Daniels
2. 2.DevOpsfor Developers, Michael Hüttermann



