

**Faculty of Sciences  
Department of Computer Science  
BS Information Technology**

**Vision**

The department of computer science (UAF) envisions to serve Nationally & Internationally through quality education, research, and outreach in the field of Computer Science to address real world challenges.

**Mission**

1. To provide state-of-the-art knowledge & skills in different computing domains to the graduates through high quality teaching.
2. To execute cutting-edge research in computing areas mapped with the National & International needs.
3. To establish an effective interface with the industry & stakeholders to facilitate the graduates in developing their understanding of existing challenges & job market.
4. To impart a sense of societal, ethical & professional responsibility in the graduates

**Program Educational Objectives (PEOs)**

After 3-5 years of graduation, the graduates will be able to;

1. Provide services in professional career by acquiring knowledge in scientific, computing and IT Administration.
2. Apply, analyze, design, optimize to formulate and solve Information Technology and Multidisciplinary problems.
3. Use their skills in ethical & professional manner to raise the satisfaction level of stakeholders and to take up higher studies, research & development, and other creative efforts in science & technology.

**Program Learning Outcomes (PLOs)**

**Knowledge:** Apply the knowledge of information technologies, computer science, engineering fundamentals, developing, and administrating IT infrastructure to solve the simple to complex industrial problems.

**Problem Analysis:** Identify, formulate, review research literature, and analyze complex computing problems reaching substantiated conclusions using first principles of mathematics, programming, and engineering sciences.

**Design/Development of Solutions:** Design software solutions for complex problems and design system components or processes that meet the specified needs with appropriate consideration.

**Investigation:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**Modern Tool Usage:** Create, select, and apply appropriate techniques, modern IT tools and programming languages for prediction and modeling of complex problems with an understanding of the limitations.

**The Scientist and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues, and the consequent responsibilities relevant to the professional engineering practice.

**Environment and Sustainability:** An ability to understand the impact of professional solutions in societal and environmental contexts and demonstrate knowledge of, and need for, sustainable development.

**Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the software development and management practices.

**Individual and Teamwork:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

**Communication:** Communicate effectively in a variety of professional contexts using written and/or oral skills.

**Project Management:** Demonstrate knowledge and understanding of the information technology management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**Lifelong Learning:** Recognize the need for IT solutions, software, and tools in different fields of life, and have the preparation and ability to implement solutions and self-paced knowledge upgradation.

<b>SCHEME OF STUDIES BS (INFORMATION TECHNOLOGY)</b> <b>4 YEARS DEGREE PROGRAM, 135 CREDIT HOURS SPREAD OVER 8 SEMESTERS</b>					
<b>EXISTING</b>			<b>PROPOSED / REVISED</b>		
<b>BS(IT) Scheme of Study</b> <b>4 years Degree Program (Bachelor of Sciences in Information Technology)</b> 133 credit hours spread over 8 semesters. Eligibility: Minimum 50% marks in Intermediate with mathematics or equivalent with mathematics			<b>BS(IT) Scheme of Study</b> <b>4 years Degree Program (Bachelor of Sciences in Information Technology)</b> Eligibility: Minimum 50% marks in Intermediate with mathematics or equivalent with mathematics		
<b>CODE</b>	<b>TITLE</b>	<b>CR. HR.</b>	<b>CODE</b>	<b>TITLE</b>	<b>CR. HR.</b>
<b>SEMESTER - 1</b>					
ENG-301	Composition and Communication Skills	3(3-0)	ENG-301	COMPOSITION AND COMMUNICATION SKILLS	3(3-0)
SSH-302	Pakistan Studies	2(2-0)	SSH-302	PAKISTAN STUDIES	2(2-0)
MATH-303	Linear Algebra	3(3-0)		REMOVED	
			MATH-404	LINEAR ALGEBRA	3(3-0)
CS-305	Introduction to Information and Communication Technologies	3(2-1)	CS-305	INTRODUCTION TO INFORMATION AND COMMUNICATION TECHNOLOGIES	3(2-1)
PY-305	Basic Electronics	3(2-1)	PY-305	BASIC ELECTRONICS	3(2-1)
BBA-410	Pakistan Economics	3(3-0)	BBA-410	ECONOMY OF PAKISTAN	3(3-0)
		<b>17</b>			<b>17</b>
<b>SEMESTER - 2</b>					
MATH-304	Multivariable Calculus	3(3-0)		REMOVED	
			MATH-305	CALCULUS-II	3(3-0)
CS-306	Digital Logic Design	3(2-1)	CS-306	DIGITAL LOGIC DESIGN	3(2-1)
CS-308	Programming Fundamentals	3(2-1)	CS-308	PROGRAMMING FUNDAMENTALS	3(2-1)
IS-401 / SSH-402	Islamic Studies / Ethics	3(3-0)	IS-401 / SSH-402	ISLAMIC STUDIES / ETHICS	3(3-0)
ENG-406	Advanced Exercises and Communication Skills	3(3-0)		REMOVED	
			DHL-405	CONTEMPORARY ETHICS	3(3-0)
BBA-507	Financial Management	3(2-1)		REMOVED	

			BBA-412	FINANCIAL MANAGEMENT-I	3(3-0)
		18			18
<b>SEMESTER - 3</b>					
SE-401	Software Engineering	3(2-1)	SE-401	SOFTWARE ENGINEERING	3(2-1)
			IS-403	روحانيت	1(1-0)
CS-405	Discrete Structures	3(3-0)	CS-405	DISCRETE STRUCTURES	3(3-0)
SSH-405	Current Affairs	3(3-0)	SSH-405	CURRENT AFFAIRS	3(3-0)
CS-407	Operating Systems	3(2-1)	CS-407	OPERATING SYSTEMS	3(2-1)
CS-409	Object Oriented Programming	3(2-1)	CS-409	OBJECT ORIENTED PROGRAMMING	3(2-1)
MATH-501	Differential Equations	3(3-0)	REMOVED		
			MATH-503	ORDINARY DIFFERENTIAL EQUATIONS	3(3-0)
		18			19
<b>SEMESTER - 4</b>					
STAT-402	Statistics and Probability	3(2-1)	STAT-402	STATISTICS AND PROBABILITY	3(2-1)
			IS-402	ترجمہ قرآن	1(1-0)
CS-406	Data Communication and Networks	3(2-1)	CS-406	DATA COMMUNICATIONS AND NETWORKS	3(2-1)
CS-408	Database Systems	3(2-1)	CS-408	DATABASE SYSTEMS	3(2-1)
CS-410	Data Structure and Algorithms	3(2-1)	CS-410	DATA STRUCTURES AND ALGORITHMS	3(2-1)
IT-412	Visual Programming	3(2-1)	IT-412	VISUAL PROGRAMMING	3(2-1)
MATH-513	Operations Research	3(3-0)	MATH-513	OPERATIONS RESEARCH	3(3-0)
		18			19
<b>SEMESTER - 5</b>					
IT-503	Information Technology Infrastructure	3(2-1)	IT-503	INFORMATION TECHNOLOGY INFRASTRUCTURE	3(2-1)
IT-505	Machine Learning	3(2-1)	IT-505	MACHINE LEARNING	3(2-1)
IT-507	Database Administration and Management	3(2-1)	IT-507	DATABASE ADMINISTRATION AND MANAGEMENT	3(2-1)
IT-509	Digital Signal Processing	3(2-1)	IT-509	DIGITAL SIGNAL PROCESSING	3(2-1)
IT-511	Artificial Intelligence	3(2-1)	IT-511	ARTIFICIAL INTELLIGENCE	3(2-1)
IT-513	Web Programming	3(2-1)	IT-513	WEB PROGRAMMING	3(2-1)
		18			18
<b>SEMESTER - 6</b>					
IT-502	Data Encryption and Security	3(2-1)	IT-502	DATA ENCRYPTION AND SECURITY	3(2-1)
IT-504	Digital Image Processing	3(2-1)	IT-504	DIGITAL IMAGE PROCESSING	3(2-1)

IT-506	Big Data Analytics	3(2-1)	IT-506	BIG DATA ANALYTICS	3(2-1)
IT-508	Cloud Computing	3(2-1)	IT-508	CLOUD COMPUTING	3(2-1)
IT-510	Internet of Things	3(2-1)	IT-510	INTERNET OF THINGS	3(2-1)
IT-512	Mobile Application Development	3(2-1)	IT-512	MOBILE APPLICATION DEVELOPMENT	3(2-1)
		<b>18</b>			<b>18</b>

### SEMESTER - 7

BBA-510	Introduction To HRM	3(3-0)	REMOVED		
			BBA-409	INTRODUCTION TO HUMAN RESOURCE MANAGEMENT	3(3-0)
IT-601	Human Computer Interaction	3(2-1)	IT-601	HUMAN COMPUTER INTERACTION	3(2-1)
			BBA-603	ENTREPRENEURSHIP	3(3-0)
IT-603	System and Network Administration	3(2-1)	IT-603	SYSTEM AND NETWORK ADMINISTRATION	3(2-1)
IT-605	Information Technology Project Management	3(2-1)	IT-605	INFORMATION TECHNOLOGY PROJECT MANAGEMENT	3(2-1)
IT-607	Next Generation Networks	3(2-1)	IT-607	NEXT GENERATION NETWORKS	3(2-1)
IT-609	Research Report	2(0-2)	IT-609	FINAL YEAR PROJECT-I	2(0-2)
		<b>17</b>			<b>20</b>

### SEMESTER - 8

BBA-603	Entrepreneurship	3(3-0)	SHIFTED TO 7 <sup>TH</sup> SEMESTER		
IT-608	Internship	2(0-2)	IT-608	INTERNSHIP	2(0-2)
IT-610	Final Year Project	4(0-4)	IT-610	FINAL YEAR PROJECT-II	4(0-4)
		<b>9</b>			<b>6</b>

**SCHEME OF STUDIES BS (INFORMATION TECHNOLOGY)**  
4 YEARS DEGREE PROGRAM, 135 CREDIT HOURS SPREAD OVER 8 SEMESTERS

**BS(IT) Scheme of Study**  
**4 years Degree Program (Bachelor of Sciences in Information Technology)**  
Eligibility: Minimum 50% marks in Intermediate with mathematics or equivalent with mathematics

CODE	TITLE	CR. HR.
<b>SEMESTER - 1</b>		
ENG-301	COMPOSITION AND COMMUNICATION SKILLS	3(3-0)
SSH-302	PAKISTAN STUDIES	2(2-0)
MATH-404	LINEAR ALGEBRA	3(3-0)
CS-305	INTRODUCTION TO INFORMATION AND COMMUNICATION TECHNOLOGIES	3(2-1)
PY-305	BASIC ELECTRONICS	3(2-1)
BBA-410	ECONOMY OF PAKISTAN	3(3-0)
<b>SEMESTER - 2</b>		
MATH-305	CALCULUS-II	3(3-0)
CS-306	DIGITAL LOGIC DESIGN	3(2-1)
CS-308	PROGRAMMING FUNDAMENTALS	3(2-1)
IS-401 / SSH-402	ISLAMIC STUDIES / ETHICS	3(3-0)
DHL-405	CONTEMPORARY ETHICS	3(3-0)
BBA-412	FINANCIAL MANAGEMENT-1	3(3-0)
<b>SEMESTER - 3</b>		
SE-401	SOFTWARE ENGINEERING	3(2-1)
IS-403	روحانیت	1(1-0)
CS-405	DISCRETE STRUCTURES	3(3-0)
SSH-405	CURRENT AFFAIRS	3(3-0)
CS-407	OPERATING SYSTEMS	3(2-1)
CS-409	OBJECT ORIENTED PROGRAMMING	3(2-1)
MATH-503	ORDINARY DIFFERENTIAL EQUATIONS	3(3-0)

		19
SEMESTER - 4		
STAT-402	STATISTICS AND PROBABILITY	3(2-1)
IS-402	ترجمہ قرآن	1(1-0)
CS-406	DATA COMMUNICATIONS AND NETWORKS	3(2-1)
CS-408	DATABASE SYSTEMS	3(2-1)
CS-410	DATA STRUCTURES AND ALGORITHMS	3(2-1)
IT-412	VISUAL PROGRAMMING	3(2-1)
MATH-513	OPERATIONS RESEARCH	3(3-0)
		19
SEMESTER - 5		
IT-503	INFORMATION TECHNOLOGY INFRASTRUCTURE	3(2-1)
IT-505	MACHINE LEARNING	3(2-1)
IT-507	DATABASE ADMINISTRATION AND MANAGEMENT	3(2-1)
IT-509	DIGITAL SIGNAL PROCESSING	3(2-1)
IT-511	ARTIFICIAL INTELLIGENCE	3(2-1)
IT-513	WEB PROGRAMMING	3(2-1)
		18
SEMESTER - 6		
IT-502	DATA ENCRYPTION AND SECURITY	3(2-1)
IT-504	DIGITAL IMAGE PROCESSING	3(2-1)
IT-506	BIG DATA ANALYTICS	3(2-1)
IT-508	CLOUD COMPUTING	3(2-1)
IT-510	INTERNET OF THINGS	3(2-1)
IT-512	MOBILE APPLICATION DEVELOPMENT	3(2-1)
		18
SEMESTER - 7		
BBA-409	INTRODUCTION TO HUMAN RESOURCE MANAGEMENT	3(3-0)
IT-601	HUMAN COMPUTER INTERACTION	3(2-1)
BBA-603	ENTREPRENEURSHIP	3(3-0)
IT-603	SYSTEM AND NETWORK ADMINISTRATION	3(2-1)
IT-605	INFORMATION TECHNOLOGY PROJECT MANAGEMENT	3(2-1)
IT-607	NEXT GENERATION NETWORKS	3(2-1)
IT-609	FINAL YEAR PROJECT-I	2(0-2)
		20

SEMESTER - 8			
IT-608	INTERNSHIP		2(0-2)
IT-610	FINAL YEAR PROJECT-II		4(0-4)
			6



**Revised Scheme of Studies for BS Information Technology 2022**

Sr.#	EXISTING			PROPOSED		
1	CS-305	Introduction to Information and Communication Technologies	3(2-1)	CS-305	INTRODUCTION TO INFORMATION AND COMMUNICATION TECHNOLOGIES	3(2-1)
2	CS-306	Digital Logic Design	3(2-1)	CS-306	DIGITAL LOGIC DESIGN	3(2-1)
3	CS-308	Programming Fundamentals	3(2-1)	CS-308	PROGRAMMING FUNDAMENTALS	3(2-1)
4	SE-401	Software Engineering	3(2-1)	SE-401	SOFTWARE ENGINEERING	3(2-1)
5	CS-405	Discrete Structures	3(3-0)	CS-405	DISCRETE STRUCTURES	3(3-0)
6	CS-407	Operating Systems	3(2-1)	CS-407	OPERATING SYSTEMS	3(2-1)
7	CS-409	Object Oriented Programming	3(2-1)	CS-409	OBJECT ORIENTED PROGRAMMING	3(2-1)
8	CS-406	Data Communication and Networks	3(2-1)	CS-406	DATA COMMUNICATIONS AND NETWORKS	3(2-1)
9	CS-408	Database Systems	3(2-1)	CS-408	DATABASE SYSTEMS	3(2-1)
10	CS-410	Data Structure and Algorithms	3(2-1)	CS-410	DATA STRUCTURES AND ALGORITHMS	3(2-1)
11	IT-412	Visual Programming	3(2-1)	IT-412	VISUAL PROGRAMMING	3(2-1)
12	IT-503	Information Technology Infrastructure	3(2-1)	IT-503	INFORMATION TECHNOLOGY INFRASTRUCTURE	3(2-1)
13	IT-505	Machine Learning	3(2-1)	IT-505	MACHINE LEARNING	3(2-1)
14	IT-507	Database Administration and Management	3(2-1)	IT-507	DATABASE ADMINISTRATION AND MANAGEMENT	3(2-1)
15	IT-509	Digital Signal Processing	3(2-1)	IT-509	DIGITAL SIGNAL PROCESSING	3(2-1)
16	IT-511	Artificial Intelligence	3(2-1)	IT-511	ARTIFICIAL INTELLIGENCE	3(2-1)
17	IT-513	Web Programming	3(2-1)	IT-513	WEB PROGRAMMING	3(2-1)
18	IT-502	Data Encryption and Security	3(2-1)	IT-502	DATA ENCRYPTION AND SECURITY	3(2-1)
19	IT-504	Digital Image Processing	3(2-1)	IT-504	DIGITAL IMAGE PROCESSING	3(2-1)
20	IT-506	Big Data Analytics	3(2-1)	IT-506	BIG DATA ANALYTICS	3(2-1)
21	IT-508	Cloud Computing	3(2-1)	IT-508	CLOUD COMPUTING	3(2-1)
22	IT-510	Internet of Things	3(2-1)	IT-510	INTERNET OF THINGS	3(2-1)
23	IT-512	Mobile Application Development	3(2-1)	IT-512	MOBILE APPLICATION DEVELOPMENT	3(2-1)
24	IT-601	Human Computer Interaction	3(2-1)	IT-601	HUMAN COMPUTER INTERACTION	3(2-1)
25	IT-603	System and Network Administration	3(2-1)	IT-603	SYSTEM AND NETWORK ADMINISTRATION	3(2-1)
26	IT-605	Information Technology Project Management	3(2-1)	IT-605	INFORMATION TECHNOLOGY PROJECT MANAGEMENT	3(2-1)
27	IT-607	Next Generation Networks	3(2-1)	IT-607	NEXT GENERATION NETWORKS	3(2-1)
28	IT-609	Research Report	2(0-2)	IT-609	FINAL YEAR PROJECT-I	2(0-2)
29	IT-608	Internship	2(0-2)	IT-608	INTERNSHIP	2(0-2)
30	IT-610	Final Year Project	4(0-4)	IT-610	FINAL YEAR PROJECT-II	4(0-4)

## COURSE CONTENTS

EXISTING			PROPOSED		
CS-305	Introduction to Information and Communication Technologies	3(2-1)	CS-305	INTRODUCTION TO INFORMATION AND COMMUNICATION TECHNOLOGIES	3(2-1)
<b>Objectives:</b> The course is designed to familiarize students with the basics computer and information systems. It also emphasizes on understanding basics of DMBS, Networks, Programming, and security.			<b>Course Learning Outcomes</b> By the end of this course, students will be able to: <ol style="list-style-type: none"> <li>1. Describe the usage of computers and why computers are essential components in business and society</li> <li>2. Utilize the Internet Web resources and evaluate on-line e-business system.</li> <li>3. Solve common business problems using appropriate Information Technology applications and systems.</li> <li>4. Identify categories of programs, system software and applications. Organize and work with files and folders.</li> <li>5. Describe various types of networks, network standards and communication software.</li> </ol>		
<b>Theory:</b> Introduction; History of Computer systems; Information system, Systems development, Organization and information systems; Competitive advantage; Hardware and software; Hardware components; Processing and memory devices; Secondary storage and input and output devices, Computer System types; Overview of software: Systems and application software, Programming languages; Issues and trends; Data management; Data modeling and the relational database model; Database management systems; Database applications; Overview of telecommunication and networks; Use and functions of the internet; Introduction to electronic commerce and e-business; Wireless technologies; Computer networks and Internet access; Mobile computing commerce; IT security and other issues.			<b>Theory</b> Introduction; History of Computer systems; Information system, systems development, organization and information systems; Competitive advantage; Hardware and software; Hardware components; Processing and memory devices; Secondary storage and input and output devices, computer system types; Overview of software: systems and application software, programming languages; Issues and trends; Data management; Data modeling and the relational database model; Database management systems; Database applications; Overview of telecommunication and networks; Use and functions of the internet; Introduction to electronic commerce and e-business; Wireless technologies; Computer networks and Internet access; Mobile computing commerce; IT security and other issues.		
<b>Practical:</b> Working with operating system: windows, working with MS-word: creating files, editing, formatting, layout, working with MS-excel: creating worksheet, applying formula, Math and Stat Functions; working with MS power point: creating power point slides, working with animations; Internet, use of different search engines: Searching and surfing; Sending and receiving mails and WWW.			<b>Practical</b> Working with operating system: windows, working with MS-word: creating files, editing, formatting, layout, working with MS-excel: creating worksheet, applying formula, Math and Stat Functions; working with MS power point: creating power point slides, working with animations; Internet, use of different search engines: Searching and surfing; Sending and receiving mails and WWW.		
<b>Suggested Readings:</b> <ol style="list-style-type: none"> <li>1. Murray, K. 2017. First Look, Microsoft Office System. Prentice Hall, New Delhi, India.</li> <li>2. Sara E.H., S.H. Clifford and S.C. Sawyer. 2016. Computers, Communications and Information: A User's Introduction. McGraw Hill, USA.</li> <li>3. Turban, E., R.K. Rainer and R. Potter. 2017. Introduction to Information Technology. John Wiley &amp; Sons, Inc., USA.</li> <li>4. Saeed, I. T. Mahmood and A. Raza. 2016. The Concept of Information technology. Kitab Markaz Faisalabad, Pakistan.</li> </ol>			<b>Suggested Readings</b> <ol style="list-style-type: none"> <li>1. Murray, K. 2019. First Look, Microsoft Office System. Prentice Hall, New Delhi, India.</li> </ol>		

<p>5. Stair, R.R. 2017. Fundamentals of Information Systems. Thomson Course Technology, USA.</p> <p>Turban, E., R.K. Rainer and R. Potter. 2016. Introduction to Information Technology. John Wiley &amp; Sons, Inc., USA.</p>	<p>2. Sara E.H., S.H. Clifford and S.C. Sawyer. 2016. Computers, Communications and Information: A User's Introduction. McGraw Hill, USA.</p> <p>3. Turban, E., R.K. Rainer and R. Potter. 2017. Introduction to Information Technology. John Wiley &amp; Sons, Inc., USA.</p> <p>4. Saeed, I. T. Mahmood and A. Raza. 2016. The Concept of Information technology. Kitab Markaz Faisalabad, Pakistan.</p> <p>5. Stair, R.R. 2017. Fundamentals of Information Systems. Thomson Course Technology, USA.</p> <p>6. Turban, E., R.K. Rainer and R. Potter. 2016. Introduction to Information Technology. John Wiley &amp; Sons, Inc., USA.</p>
<b>CS-306</b> <b>Digital Logic Design</b> <b>3(2-1)</b>	<b>CS-306</b> <b>DIGITAL LOGIC DESIGN</b> <b>3(2-1)</b>
<p><b>Theory:</b> Number Systems: digital systems, binary numbers, number-base conversions, octal and hexadecimal numbers, complements of numbers, signed binary numbers, binary codes, binary logic; Fundamentals of Boolean algebra: Boolean functions, digital logic gates, Integrated circuits; Simplification Methods (K-Map, Quinn Mc-Cluskey method); Nand and nor implementation, exclusive-OR function; Combinational logic circuits: binary adder and subtract, decimal adder, binary multiplier, decoders, encoders, multiplexers; Adders; comparators; encoders and decoders; Multiplexers and de-multiplexers; Flip Flops and Latches, Asynchronous and Synchronous circuits, Shift Registers, Triggered devices &amp; its types., Memory Elements, State Machines.</p> <p><b>Practical:</b> To design and implementation of the operation of logic gates, half adder, full adder and half subtract circuits, segment display, common BCD to 7-segment decoder using IC 74LS47, design a single-line digital communication circuit using a multiplexer and decoder, operation of RS, D, JK and T flip flops working with Programmable Logic Devices (CPLD, FPGA); working with circuits using tools such as Verilog HDL/VHDL, MultiSim.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Gajski, D. 2017. Principles of Digital Design. McGraw Hill, USA.</li> <li>2. Guy, E. 2019. Digital Logic Design. Cambridge University Press, UK.</li> <li>3. Morris, M. 2017. Digital Design: With an Introduction to the Verilog HDL, VHDL, and System Verilog. 6th Ed. Pearson, USA.</li> <li>4. Rafiquzzaman, M. 2016. Fundamentals of Digital Logic and Microcontrollers, John Wiley &amp; Sons, Inc., USA.</li> <li>5. Sarah, H. 2015. Digital Design and Computer Architecture. Morgan Kaufmann, USA.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Investigate, design and analyze combinational logic circuits</li> <li>2. Analyze small-scale combinational and sequential digital circuits</li> <li>3. Design and analyze sequential logic circuits</li> <li>4. Describe the basic software tools for the design and implementation of digital circuits and systems</li> <li>5. Demonstrate the acquired knowledge to apply techniques related to the design and analysis of digital electronic circuits including Boolean algebra and multi-variable Karnaugh map methods</li> </ol> <p><b>Theory</b> Number Systems: digital systems, binary numbers, number-base conversions, octal and hexadecimal numbers, complements of numbers, signed binary numbers, binary codes, binary logic; Fundamentals of Boolean algebra: Boolean functions, digital logic gates, Integrated circuits; Simplification Methods (K-Map, Quinn Mc-Cluskey method); Nand and nor implementation, exclusive-OR function; Combinational logic circuits: binary adder and subtract, decimal adder, binary multiplier, decoders, encoders, multiplexers; Adders; comparators; encoders and decoders; Multiplexers and de-multiplexers; Flip Flops and Latches, Asynchronous and Synchronous circuits, Shift Registers, Triggered devices &amp; its types., Memory Elements, State Machines.</p> <p><b>Practical</b> To design and implementation of the operation of logic gates, half adder, full adder and half subtract circuits, segment display, common BCD to 7-segment decoder using IC 74LS47, design a single-line digital communication circuit using a multiplexer and decoder, operation</p>

			of RS, D, JK and T flip flops working with Programmable Logic Devices (CPLD, FPGA); working with circuits using tools such as Verilog HDL/VHDL, MultiSim. <b>Suggested Readings</b> 1. Gajski, D. 2017. Principles of Digital Design. McGraw Hill, USA. 2. Guy, E. 2019. Digital Logic Design. Cambridge University Press, UK. 3. Morris, M. 2017. Digital Design: With an Introduction to the Verilog HDL, VHDL and System Verilog. 6th Ed. Pearson, USA. 4. Rafiquzzaman, M. 2016. Fundamentals of Digital Logic and Microcontrollers, John Wiley & Sons, Inc., USA. 5. Sarah, H. 2015. Digital Design and Computer Architecture. Morgan Kaufmann, USA.
<b>CS-308</b>	<b>Programming Fundamentals</b>	<b>3(2-1)</b>	<b>CS-308 PROGRAMMING FUNDAMENTALS 3(2-1)</b>
<b>Objectives:</b> The course aims to focus on fundamental concepts of programming, analysis, and software development. <b>Theory:</b> Introduction to problem solving; a brief review of Von-Neumann architecture; introduction to programming; role of compiler and linker; introduction to algorithms; basic data types and variables; input/output constructs; arithmetic; comparison and logical operators; conditional statements and execution flow for conditional statements; repetitive statements and execution flow for repetitive statements; lists and their memory organization, multi-dimensional lists; introduction to modular programming; function definition and calling; stack rolling and unrolling; string and string operations; pointers/references; static and dynamic memory allocation; File I/O operations. <b>Practical:</b> Design simple structure of program; solving different programs using: Sequential structure (If, if-else, switch) and Iterative structure (while, do-while, for); solving different programs using one dimensional and multidimensional arrays; use of function in programs; writing user defined functions; calling functions with parameters; programming using the pointers; pointers and arrays; Use of String functions. <b>Suggested Readings:</b> 1. Deitel, P. and H. Deitel. 2017. C++ How to Program. 10th Ed., Prentice Hall, USA. 2. Hanly, J.R. and E.B. Koffman. 2017. Problem Solving and Program Design in C++, Addison-Wesley. 3. Harvey, M.D. and P.J. Deitel. 2016. C++ How to Program. Prentice Hall Publications.			<b>Course Learning Outcomes</b> By the end of this course, students will be able to: 1. Illustrate a disciplined approach to Problem solving methods and algorithm development. The aim is to teach the syntax and vocabulary of a modern programming language like C++ 2. Write fundamental concepts of structured programming along with problem solving techniques and analytical thinking. 3. Analyze problem requirements to recognize what type of data and processes are involved in solution. 4. Restate basic concepts to alter and expand short programs and algorithms. Writing pseudo-codes and simple program development using standard control structures and functions. 5. Design module-based approach to gratify those requirements and organize program code for implementation.  <b>Theory</b> Introduction to problem solving; a brief review of Von-Neumann architecture; introduction to programming; role of compiler and linker; introduction to algorithms; basic data types and variables; input/output constructs; arithmetic; comparison and logical operators; conditional statements and execution flow for conditional statements; repetitive statements and execution flow for repetitive statements; lists and their memory organization, multi-dimensional lists; introduction to modular programming; function definition and calling; stack rolling and unrolling; string and string operations; pointers/references; static and dynamic memory allocation; File I/O operations.  <b>Practical</b>

<ol style="list-style-type: none"> <li>4. Lafore R. 2015. Object Oriented Programming in C++ 4<sup>th</sup> edition. Sams Publishing, USA.</li> <li>5. Mustafa, T. and A.R. Sattar. 2015. Object oriented programming using C++. IT Series, Pakistan.</li> <li>6. Zaik D. 2016. An introduction to programming with C++. Cengage Learning, Inc, USA.</li> </ol>	<p>Design simple structure of program; solving different programs using: Sequential structure (If, if-else, switch) and Iterative structure (while, do-while, for); solving different programs using one dimensional and multidimensional arrays; use of function in programs; writing user defined functions; calling functions with parameters; programming using the pointers; pointers and arrays; Use of String functions.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Deitel, P. and H. Deitel. 2019. C++ How to Program. 10th Ed., Prentice Hall, USA.</li> <li>2. Hanly, J.R. and E.B. Koffman. 2017. Problem Solving and Program Design in C++, Addison-Wesley.</li> <li>3. Lafore R. 2015. Object Oriented Programming in C++ 4<sup>th</sup> edition. Sams Publishing, USA.</li> <li>4. Mustafa, T. and A.R. Sattar. 2015. Object oriented programming using C++. IT Series, Pakistan.</li> <li>5. Zaik D. 2016. An introduction to programming with C++. Cengage Learning, Inc, USA.</li> </ol>
<p><b>SE-401</b>                      <b>Software Engineering</b>                      <b>3(2-1)</b></p>	<p><b>SE-401</b>                      <b>SOFTWARE ENGINEERING</b>                      <b>3(2-1)</b></p>
<p><b>Objectives:</b> To study various software development models and phases of software development life cycle.</p> <p><b>Theory:</b> Introduction to software; Software characteristics; Software components; Software applications; Software engineering as a layered technology; Software process models: Generic Process model, process assessment and improvement, prescriptive process models, specialized process models, personal and team process models; Agile Development: Agile process, extreme programming, advanced agile process models; Requirement Engineering: building the requirement model, negotiating requirements, validating; 4GLs; Software development architectures; Introduction to CASE tools; Design and implementation: Design process, design concepts, design model, Architectural design, component level design; UML diagrams; Software testing and quality assurance: formal approaches, software reliability, software testing strategies; Software evolution, Project management and project planning, configuration management: SCM repository, SCM process.</p> <p><b>Practical:</b> Development of DFD, data dictionary, E-R diagram; structured chart for the project; draw various UML diagrams; To illustrate the use of class diagrams; To draw an activity diagram and use case diagram for some software from real life like library management/campus</p>	<p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.</li> <li>2. To provide an idea of using various process models in the software industry according to given circumstances.</li> <li>3. To study various software development models and phases of software development life cycle.</li> <li>4. Basic knowledge and understanding of the analysis and design of complex systems.</li> <li>5. Ability to apply software engineering principles and techniques</li> </ol> <p><b>Theory</b> Introduction to software; Software characteristics; Software components; Software applications; Software engineering as a layered technology; Software process models: Generic Process model, process assessment and improvement, prescriptive process models, specialized process models, personal and team process models; Agile Development: Agile process, extreme programming, advanced agile process models; Requirement Engineering: building the requirement model, negotiating requirements, validating; 4GLs; Software development architectures; Introduction to CASE tools; Design and implementation: Design process, design concepts, design model, Architectural design, component level design; UML</p>

<p>management system etc; Draw Object Diagram for any software; Development of State Transition Diagram; Draw ER Diagram for some systems like Hospital Management System.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Jeffrey L.W, D.B. Lonnie and C.D. Kevin. 2016. System Analysis and Design Methods. McGraw Hill, Inc., USA.</li> <li>2. Pressman, R.S. 2015. Software Engineering A Practitioner's Approach. McGraw Hill, Inc., USA.</li> <li>3. Ralph, J. 2016. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, USA.</li> <li>4. Robert C. 2018. Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall, USA.</li> </ol> <p>Robert C. 2017. Clean Architecture: A Craftsman's Guide to Software Structure and Design, Prentice Hall, USA.</p>	<p>diagrams; Software testing and quality assurance: formal approaches, software reliability, software testing strategies; Software evolution, Project management and project planning, configuration management: SCM repository, SCM process.</p> <p><b>Practical</b></p> <p>Development of DFD, data dictionary, E-R diagram; structured chart for the project; draw various UML diagrams; To illustrate the use of class diagrams; To draw an activity diagram and use case diagram for some software from real life like library management/campus management system etc; Draw Object Diagram for any software; Development of State Transition Diagram; Draw ER Diagram for some systems like Hospital Management System.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Jeffrey L.W, D.B. Lonnie, and C.D. Kevin. 2016. System Analysis and Design Methods. McGraw Hill, Inc., USA.</li> <li>2. Pressman, R.S. 2015. Software Engineering A Practitioner's Approach. McGraw Hill, Inc., USA.</li> <li>3. Ralph, J. 2016. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, USA.</li> <li>4. Robert C. 2018. Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall, USA.</li> <li>5. Robert C. 2017. Clean Architecture: A Craftsman's Guide to Software Structure and Design, Prentice Hall, USA.</li> </ol>
<p><b>CS-405</b>                      <b>Discrete Structures</b>                      <b>3(3-0)</b></p>	<p><b>CS-405</b>                      <b>DISCRETE STRUCTURES</b>                      <b>3(3-0)</b></p>
<p><b>Theory:</b></p> <p>The Foundations: Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference; Basic Structures: Sets, Functions, Sequences, Sums, and Matrices; Algorithms: The Growth of Functions, Complexity of Algorithms; Number Theory and Cryptography: Divisibility and Modular Arithmetic, Integer Representations and Algorithms, Primes and Greatest Common Divisors, Solving Congruences, Applications of Congruences; Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness; Counting: The Basics of Counting, The Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations; Discrete Probability; Relations; Graphs; Trees; Boolean Algebra; Modeling Computation.</p> <p><b>Suggested Readings:</b></p>	<p><b>Course Learning Outcomes</b></p> <p>By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the understanding of Logic Sets and Functions.</li> <li>2. Use mathematical reasoning techniques including induction and recursion.</li> <li>3. Apply counting techniques to the representation and characterization of relational concepts.</li> <li>4. Explain how graph and tree concepts are used to solve problems arising in the computer science.</li> <li>5. Implement formal reasoning skills.</li> </ol> <p><b>Theory</b></p> <p>The Foundations: Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference; Basic Structures: Sets, Functions, Sequences, Sums, and Matrices; Algorithms: The Growth of Functions, Complexity of Algorithms; Number Theory and Cryptography:</p>

<ol style="list-style-type: none"> <li>1. Epp, S. S. 2019. Application of Discrete Applied Mathematics, 5th Ed. Cengage, USA.</li> <li>2. Judith, L. 2015. Mathematical Structure for Computer Science. McGraw Hill. Inc., USA.</li> <li>3. Kenneth, H. 2018. Connect Access Card for Discrete Mathematics and Its Applications. McGraw-Hill, USA.</li> <li>4. Ralph, P. 2016. Discrete and Combinatorial Mathematics: An Applied Introduction. Pearson, USA.</li> </ol> <p>Winifred, G. 2015. Logic and Discrete Mathematics: A Computer Science Perspective. Prentice Hall, USA.</p>	<p>Divisibility and Modular Arithmetic, Integer Representations and Algorithms, Primes and Greatest Common Divisors, Solving Congruences, Applications of Congruences; Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness; Counting: The Basics of Counting, The Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations; Discrete Probability; Relations; Graphs; Trees; Boolean Algebra; Modeling Computation.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Epp, S. S. 2019. Application of Discrete Applied Mathematics, 5th Ed. Cengage, USA.</li> <li>2. Judith, L. 2015. Mathematical Structure for Computer Science. McGraw Hill. Inc., USA.</li> <li>3. Kenneth, H. 2018. Connect Access Card for Discrete Mathematics and Its Applications. McGraw-Hill, USA.</li> <li>4. Ralph, P. 2016. Discrete and Combinatorial Mathematics: An Applied Introduction. Pearson, USA.</li> <li>5. Winifred, G. 2015. Logic and Discrete Mathematics: A Computer Science Perspective. Prentice Hall, USA.</li> <li>6. Richard Johnsonbaugh, "Discrete Mathematics", 8th Edition, Pearson Education Asia, 2018.</li> </ol>
<p><b>CS-407</b> <b>Operating Systems</b> <b>3(2-1)</b></p>	<p><b>CS-407</b> <b>OPERATING SYSTEMS</b> <b>3(2-1)</b></p>
<p><b>Objectives:</b> This course provides the overview of computer system and the operating system, the concepts of process management, memory management, storage management, protection and security issues, and distributed systems.</p> <p><b>Theory:</b> Overview of various types of systems: Simple batch systems, Multi-programmed batch systems, Time-sharing systems, Personal computer systems, Parallel systems, Distributed systems and Real-time System; Interrupts and its types; What is Kernel and its types; System Calls; Process Management: Concept of a process, Process States, Process Control Block, Process Scheduling and Operations on Processes; Thread: Introduction, Advantages, and types of thread; CPU Scheduling: Multi Programming, Scheduling Objectives, Scheduler, Dispatcher, Preemptive and non- Preemptive scheduling and Scheduling Criteria; Scheduling Algorithms; Deadlock: Introduction, Resources, Deadlock Characterization,</p>	<p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Compare the functionality of different Operating Systems Structures.</li> <li>2. Describe issues of Process Management including process structure, synchronization, scheduling and communication.</li> <li>3. Demonstrate memory management issues including advance techniques of paging, segmentation and virtual memory.</li> <li>4. Explain the operation of various File Management Algorithms.</li> <li>5. Explain the issues related to I/O Sub-systems, Threats and specialized operating systems.</li> </ol> <p><b>Theory</b> Overview of various types of systems: Simple batch systems, Multi-programmed batch systems, Time-sharing systems, Personal computer systems, Parallel systems, Distributed</p>



<p>Resource Allocation Graphs, Methods for “Handling Deadlocks, Deadlock Preventions, Deadlock Avoidance, Detection and Recovery; Memory Management: Introduction, Basic Requirements, Contiguous vs Non-Contiguous Storage Allocation, Swapping, Memory Allocation Methods, Logical vs Physical Address Space and Paging; Virtual Memory: Introduction to Virtual Memory, Demand Paging, Page Replacement Algorithm, Allocation of Frames, Thrashing, Working Set and Page Size; File System Management: File Attributes, File Operations, Access Methods, Directory Structure, I/O Management: I/O Operations, Handshaking, Interrupts, Classes of Interrupts and I/O Devices; System protection; Virtual machines; Operating system security.</p> <p><b>Practical:</b> Installation of different Operating Systems (Window, Unix, Linux); Operating system commands, file creation &amp; management; Scheduling Algorithms: First In First Out (FIFO), Shortest Job First, Priority Scheduling and Round Robin; Memory Allocation methods; System Performance Optimization; Deadlock detection and recovery techniques; Creating &amp; Managing threads using Modern Programming languages; Directory Structure Management.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Abraham Silberschatz. 2018. Operating Systems Concepts, 10th Ed. Wiley, USA.</li> <li>2. Andrew S. Tanenbaum. 2016. Modern Operating Systems. 4th Ed. Pearson, USA.</li> <li>3. Hajek, D. and C, Herrera. 2019. Principles of Operating Systems. 2019 Ed. USA.</li> <li>4. Thomas W. Doepfner. 2010. Operating System in Depth: Design and Programming. 1st Ed. John Wiley &amp; Sons, USA.</li> <li>5. William Stallings. 2018. Operating Systems Internals and Design Principles, 9th Ed. Pearson, USA.</li> </ol>	<p>systems and Real-time System; Interrupts and its types; What is Kernel and its types; System Calls; Process Management: Concept of a process, Process States, Process Control Block, Process Scheduling and Operations on Processes; Thread: Introduction, Advantages, and types of thread; CPU Scheduling: Multi Programming, Scheduling Objectives, Scheduler, Dispatcher, Preemptive and non- Preemptive scheduling and Scheduling Criteria; Scheduling Algorithms; Deadlock: Introduction, Resources, Deadlock Characterization, Resource Allocation Graphs, Methods for “Handling Deadlocks, Deadlock Preventions, Deadlock Avoidance, Detection and Recovery; Memory Management: Introduction, Basic Requirements, Contiguous vs Non-Contiguous Storage Allocation, Swapping, Memory Allocation Methods, Logical vs Physical Address Space and Paging; Virtual Memory: Introduction to Virtual Memory, Demand Paging, Page Replacement Algorithm, Allocation of Frames, Thrashing, Working Set and Page Size; File System Management: File Attributes, File Operations, Access Methods, Directory Structure, I/O Management: I/O Operations, Handshaking, Interrupts, Classes of Interrupts and I/O Devices; System protection; Virtual machines; Operating system security.</p> <p><b>Practical</b> Installation of different Operating Systems (Window, Unix, Linux); Operating system commands, file creation &amp; management; Scheduling Algorithms: First In First Out (FIFO), Shortest Job First, Priority Scheduling and Round Robin; Memory Allocation methods; System Performance Optimization; Deadlock detection and recovery techniques; Creating &amp; Managing threads using Modern Programming languages; Directory Structure Management.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Andrew S. Tanenbaum. 2016. Modern Operating Systems. 4<sup>th</sup> Ed. Pearson, USA.</li> <li>2. Hajek, D. and C, Herrera. 2019. Principles of Operating Systems. 2019 Ed. USA.</li> <li>3. Silberschatz A. 2018. Operating Systems Concepts, 10<sup>th</sup> Ed. Wiley, USA.</li> <li>4. Thomas W. Doepfner. 2010. Operating System in Depth: Design and Programming. 1<sup>st</sup> Ed. John Wiley &amp; Sons, USA.</li> <li>5. William Stallings. 2018. Operating Systems Internals and Design Principles, 9<sup>th</sup> Ed. Pearson, USA.</li> </ol>
<p><b>CS-409</b>                      <b>Object Oriented Programming</b>                      <b>3(2-1)</b></p> <p><b>Theory:</b> History and advantages of object oriented design; Introduction to object oriented programming concepts: classes, objects, data encapsulation, constructors, destructors, access modifiers, static data members &amp; functions; Function overloading; Operator overloading; this Pointer; Constant data members &amp; Static Variables; Array of objects; new</p>	<p><b>CS-409</b>                      <b>OBJECT ORIENTED PROGRAMMING</b>                      <b>3(2-1)</b></p> <p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Explain knowledge of underlying concepts of object-oriented paradigm.</li> <li>2. Describe an understanding of object-oriented design artifacts and their mapping to object-oriented programming.</li> </ol>



<p>Operator; Getter and Setter; Composition: Composition, Aggregation and Friend Functions; Identification of classes and their relationships; inheritance; multiple inheritance; polymorphism; String manipulation; Abstract classes and interfaces; Class templates; Standard template Library; Object streams, data and object serialization using object streams; I/O and file processing; Exception handling; Event handling; Debugging strategies; Swing interactors, Graphical user interfaces; Event driven programming and Database connectivity.</p> <p><b>Practical:</b> Compiling &amp; Debugging, Coding patterns and styles; Java docs; Variables, Operators; Selection Statements, Control Statements; Looping structures, Classes; Methods; Overloading; Strings manipulation; Inheritance; Polymorphism; Graphic &amp; Events handling; I/O and file processing; Exception handling; Abstract classes; Interfaces; Exception handling; Garbage collection; Packages and Semester Project.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Deitel, P. and H. Deitel. 2017. C++ How to Program. 10<sup>th</sup> Ed., Prentice Hall, USA.</li> <li>2. Deitel, P. and H. Deitel. 2017. Java How to Program. 11<sup>th</sup> Ed., Prentice Hall. USA.</li> <li>3. Gaddis, T. and P. Sengupta. 2017. Starting out with C++: from control structures through objects. 9<sup>th</sup> Ed. Pearson, USA.</li> <li>4. Mustafa, T. and A.R. Sattar. 2017. Object Oriented Programming using C++, IT Series Publications, Pakistan.</li> <li>5. Schildt, H. 2018. Java: The Complete Reference, 11<sup>th</sup> Ed. McGraw-Hill Education Group, USA.</li> </ol>	<ol style="list-style-type: none"> <li>3. Design and implement object-oriented solutions for small systems involving single/multiple objects.</li> <li>4. Design object-oriented solutions for small systems involving multiple objects.</li> <li>5. Apply good programming style and understand the impact of style on developing and maintaining programs.</li> </ol> <p><b>Theory</b> History and advantages of object oriented design; Introduction to object oriented programming concepts: classes, objects, data encapsulation, constructors, destructors, access modifiers, static data members &amp; functions; Function overloading; Operator overloading; this Pointer; Constant data members &amp; Static Variables; Array of objects; new Operator; Getter and Setter; Composition: Composition, Aggregation and Friend Functions; Identification of classes and their relationships; inheritance; multiple inheritance; polymorphism; String manipulation; Abstract classes and interfaces; Class templates; Standard template Library; Object streams, data and object serialization using object streams; I/O and file processing; Exception handling; Event handling; Debugging strategies; Swing interactors, Graphical user interfaces; Event driven programming and Database connectivity.</p> <p><b>Practical</b> Compiling &amp; Debugging, Coding patterns and styles; Java docs; Variables, Operators; Selection Statements, Control Statements; Looping structures, Classes; Methods; Overloading; Strings manipulation; Inheritance; Polymorphism; Graphic &amp; Events handling; I/O and file processing; Exception handling; Abstract classes; Interfaces; Exception handling; Garbage collection; Packages and Semester Project.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Deitel, P. and H. Deitel. 2019. C++ How to Program. 10<sup>th</sup> Ed., Prentice Hall, USA.</li> <li>2. Deitel, P. and H. Deitel. 2017. Java How to Program. 11<sup>th</sup> Ed., Prentice Hall. USA.</li> <li>3. Gaddis, T. and P. Sengupta. 2017. Starting out with C++: from control structures through objects. 9<sup>th</sup> Ed. Pearson, USA.</li> <li>4. Mustafa, T. and A.R. Sattar. 2017. Object Oriented Programming using C++, IT Series Publications, Pakistan.</li> <li>5. Schildt, H. 2018. Java: The Complete Reference, 11<sup>th</sup> Ed. McGraw-Hill Education Group, USA.</li> </ol>
<p><b>CS-406</b>                      <b>Data Communication and Networks</b>                      <b>3(2-1)</b></p> <p><b>Theory:</b> Communications Model, Data Communication and Networking: LAN, MAN, WAN, PAN, SAN; Network topologies: star, bus, ring and tree topologies; Functions: Encapsulation,</p>	<p><b>CS-406</b>                      <b>DATA COMMUNICATIONS AND NETWORKS</b>                      <b>3(2-1)</b></p> <p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the rudiments of how computers communicate.</li> </ol>

Segmentation, Connection Control, Ordered Delivery, Flow Control, Error Control and Multiplexing; Data Transmission: Concept and Terminology, Analog and Digital Data Transmission; Transmission Medium: Guided and Un-Guided Media; Networking Types, Protocols and protocols architecture: Three Layer Model, TCP/IP Protocol Architecture, The OSI Model; Basic concepts of networking; Network topologies; layered architecture; physical layer functionality, data link layer functionality, multiple access techniques, circuit switching and packet switching, LAN technologies, wireless networks, MAC addressing, networking devices, network layer protocols, IPv4 and IPv6, IP addressing, sub netting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks.

**Practical:**

Understanding and using Network Simulators; Straight and cross cable connection; configure of access point; configure wireless routers; Subnetting; Installation of different services: DNS, DHCP; Network monitoring and analyzing tools; applying different policies; creating client/server network; shared H/W component in a network; Designing and implementation of different networks and network Topologies; Multiplexing, Quality of Service in Networks.

**Suggested Readings:**

1. Andrew S. T. 2010. Computer Networks, 5<sup>th</sup> Ed. Prentice Hall, USA.
2. Behrouz A. F. 2012. Data Communications and Networking. 5<sup>th</sup> Ed. McGraw-Hill, USA.
3. James K. and K. Ross. 2016. Computer Networking: A Top-Down Approach. 7<sup>th</sup> Ed. Pearson, USA.
4. Scott C. 2019. Networking for Beginners: Be Familiar with Computer Network Basics. USA.
5. S. William. 2013. Data and Computer Communications. 10<sup>th</sup> Ed. Pearson, USA.
6. White, R. and Banks, E. 2018. Computer Networking Problems and Solutions: An innovative approach to building resilient, modern networks. Addison-Wesley Professional, USA.

2. Explain the architecture of a number of different networks.
3. Apply the principles of protocol layering.
4. Demonstrate the modern communication systems.
5. Apply the basic aspects of packet-based protocol design and implementation.

**Theory**

Communications Model, Data Communication and Networking: LAN, MAN, WAN, PAN, SAN; Network topologies: star, bus, ring and tree topologies; Functions: Encapsulation, Segmentation, Connection Control, Ordered Delivery, Flow Control, Error Control and Multiplexing; Data Transmission: Concept and Terminology, Analog and Digital Data Transmission; Transmission Medium: Guided and Un-Guided Media; Networking Types, Protocols and protocols architecture: Three Layer Model, TCP/IP Protocol Architecture, The OSI Model; Basic concepts of networking; Network topologies; layered architecture; physical layer functionality, data link layer functionality, multiple access techniques, circuit switching and packet switching, LAN technologies, wireless networks, MAC addressing, networking devices, network layer protocols, IPv4 and IPv6, IP addressing, sub netting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks.

**Practical**

Understanding and using Network Simulators; Straight and cross cable connection; configure of access point; configure wireless routers; Subnetting; Installation of different services: DNS, DHCP; Network monitoring and analyzing tools; applying different policies; creating client/server network; shared H/W component in a network; Designing and implementation of different networks and network Topologies; Multiplexing, Quality of Service in Networks.

**Suggested Readings**

1. Andrew S. T. 2010. Computer Networks, 5<sup>th</sup> Ed. Prentice Hall, USA.
2. Behrouz A. F. 2012. Data Communications and Networking. 5<sup>th</sup> Ed. McGraw-Hill, USA.
3. James K. and K. Ross. 2016. Computer Networking: A Top-Down Approach. 7<sup>th</sup> Ed. Pearson, USA.
4. Scott C. 2019. Networking for Beginners: Be Familiar with Computer Network Basics. USA.
5. S. William. 2013. Data and Computer Communications. 10<sup>th</sup> Ed. Pearson, USA.

	6. White, R. and Banks, E. 2018. Computer Networking Problems and Solutions: An innovative approach to building resilient, modern networks. Addison-Wesley Professional, USA.
CS-408 Database Systems 3(2-1)	CS-408 DATABASE SYSTEMS 3(2-1)
<p><b>Theory:</b> Basic database concepts; Database approach vs file based system; database architecture: three level schema architecture; data independence; relational data model: attributes, schemas, tuples, domains, relation instances, keys of relations, integrity constraints; relational algebra: selection, projection, Cartesian product, types of joins; normalization: functional dependencies, normal forms; entity relationship model: entity sets, attributes, relationship, entity-relationship diagrams; Structured Query Language (SQL); Joins and sub-queries in SQL; Grouping and aggregation in SQL; concurrency control; database backup and recovery; indexes; NoSQL systems.</p> <p><b>Practical:</b> Normalization of data; writing queries using SQL; restricting and sorting the data; retrieving data from more than one tables; single row function; multiple row functions; writing sub-queries; inserting the data in database; deleting the data in database; updating the data in database; creating tables using SQL statements; implementing different types of constraints on data; creation of views; indexing.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Connolly, R. and P. Begg. 2015. Database Systems: A Practical Approach to Design, Implementation and Management. Addison-Wesley Pub. Co., USA.</li> <li>2. Coronel, C. and Morris, S. 2018. Database systems: design, implementation, &amp; management. 13<sup>th</sup> Ed. Cengage Learning. USA.</li> <li>3. Date, C. J. 2015. Database Systems. Addison Wesley Pub. Co., USA.</li> <li>4. Elmasri, R. and S. Navathe. 2016. Fundamentals of Database Systems. Addison-Wesley, USA.</li> <li>5. Mustafa, T. and A.R. Sattar. 2018. Database Management System, IT Series Publications, Pakistan.</li> <li>6. Silberschatz A., H.F. Korth and S. Sudarshan. 2019. Data base System Concepts. McGraw-Hill, USA.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Develop a database schema from database requirements</li> <li>2. Formulate SQL queries for database construction, manipulation, and retrieval</li> <li>3. Draw and Investigate Data Flow and Entity Relationship Diagrams</li> <li>4. List Relational Data Models and Relational Algebra Operators</li> <li>5. Improve the database design by normalization</li> </ol> <p><b>Theory</b> Basic database concepts; Database approach vs file based system; database architecture: three level schema architecture; data independence; relational data model: attributes, schemas, tuples, domains, relation instances, keys of relations, integrity constraints; relational algebra: selection, projection, Cartesian product, types of joins; normalization: functional dependencies, normal forms; entity relationship model: entity sets, attributes, relationship, entity-relationship diagrams; Structured Query Language (SQL); Joins and sub-queries in SQL; Grouping and aggregation in SQL; concurrency control; database backup and recovery; indexes; NoSQL systems.</p> <p><b>Practical</b> Normalization of data; writing queries using SQL; restricting and sorting the data; retrieving data from more than one tables; single row function; multiple row functions; writing sub-queries; inserting the data in database; deleting the data in database; updating the data in database; creating tables using SQL statements; implementing different types of constraints on data; creation of views; indexing.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Connolly, R. and P. Begg. 2015. Database Systems: A Practical Approach to Design, Implementation and Management. Addison-Wesley Pub. Co., USA.</li> <li>2. Coronel, C. and Morris, S. 2018. Database systems: design, implementation, &amp; management. 13<sup>th</sup> Ed. Cengage Learning. USA.</li> <li>3. Date, C. J. 2015. Database Systems. Addison Wesley Pub. Co., USA.</li> <li>4. Elmasri, R. and S. Navathe. 2016. Fundamentals of Database Systems. Addison-Wesley, USA.</li> <li>6. Mustafa, T. and A.R. Sattar. 2018. Database Management System, IT Series Publications, Pakistan.</li> </ol>

	7. Silberschatz A., H.F. Korth and S. Sudarshan. 2019. Data base System Concepts. McGraw-Hill, USA.
CS-410 Data Structure and Algorithms 3(2-1)	CS-410 DATA STRUCTURES AND ALGORITHMS 3(2-1)
<p><b>Objectives:</b> The course covers data structures such as dynamic arrays, linked lists, stacks, queues, tree, heap, disjoint sets and table and implement these in C++ and determine which structures are appropriate in various situations</p> <p><b>Theory:</b> Abstract data types, Complexity analysis; Big Oh notation; Stacks (linked lists and array implementations); Recursion and analyzing recursive algorithms; Divide and conquer algorithms; Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket); Queue: dequeuer, priority queues (linked and array implementations of queues); Linked list &amp; its various types, sorted linked list; Searching an unsorted array; Binary search for sorted arrays; Hashing and indexing; Open addressing and chaining; Trees and Tree traversals; Binary search trees, Heaps; M-way Tress, Balanced trees; Graphs: breadth-first, depth-first;, Dijkstra's Algorithm, Prim's Algorithm, Kruskal's Algorithm, shortest path; Adjacency matrix and adjacency list implementations; Memory management; Garbage collection and Dynamic Programming.</p> <p><b>Practical:</b> Implementation of Basic Data Structures: Linked lists, Stacks, Queues in modern Object-oriented Programming Languages; Implementing Sorting Algorithms; Searching Algorithms, Trees: Binary Trees, Binary Tree Manipulation, Ordered Binary Trees, Binary Search Trees, Heaps and Priority Queues, AVL Trees, B Trees, etc; Graphs and Graph Algorithms: Dijkstra's Algorithm, Prim's Algorithm, Kruskal's Algorithm, Depth-First vs Breadth-FirstSearch, Connected Components, Maximum Flow.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Bancila, M. 2017.Modern C++ Programming Cookbook, Packt Publishing, USA.</li> <li>2. Carrano, F.M. and W.J. Savitch. 2014. Data Structures and Abstractions with Java. 4th Ed. Pearson, USA.</li> <li>3. Drozdek, A. 2012. Data Structures and Algorithms in C++. Cengage Learning, USA.</li> <li>4. Goodrich, M. T. and R. Tamassia. 2016. Data structures and algorithms in Python. Wiley, India.</li> <li>5. Pai, G.A.V. 2008. Data Structures and Algorithms: Concepts, Techniques and Applications, Mc Graw Hill, USA.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the basic data structures and their implementations.</li> <li>2. Explain importance of data structures in context of writing efficient programs.</li> <li>3. Apply appropriate data structures in problem solving</li> </ol> <p><b>Theory</b> Abstract data types, Complexity analysis; Big Oh notation; Stacks (linked lists and array implementations); Recursion and analyzing recursive algorithms; Divide and conquer algorithms; Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket); Queue: dequeuer, priority queues (linked and array implementations of queues); Linked list &amp; its various types, sorted linked list; Searching an unsorted array; Binary search for sorted arrays; Hashing and indexing; Open addressing and chaining; Trees and Tree traversals; Binary search trees, Heaps; M-way Tress, Balanced trees; Graphs: breadth-first, depth-first;, Dijkstra's Algorithm, Prim's Algorithm, Kruskal's Algorithm, shortest path; Adjacency matrix and adjacency list implementations; Memory management; Garbage collection and Dynamic Programming.</p> <p><b>Practical</b> Implementation of Basic Data Structures: Linked lists, Stacks, Queues in modern Object-oriented Programming Languages; Implementing Sorting Algorithms; Searching Algorithms, Trees: Binary Trees, Binary Tree Manipulation, Ordered Binary Trees, Binary Search Trees, Heaps and Priority Queues, AVL Trees, B Trees, etc; Graphs and Graph Algorithms: Dijkstra's Algorithm, Prim's Algorithm, Kruskal's Algorithm, Depth-First vs Breadth-FirstSearch, Connected Components, Maximum Flow.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Carrano, F.M. and W.J. Savitch. 2014. Data Structures and Abstractions with Java. 4th Ed. Pearson, USA.</li> <li>2. Drozdek, A. 2012. Data Structures and Algorithms in C++. Cengage Learning, USA.</li> <li>3. Goodrich, M. T. and R. Tamassia. 2016. Data structures and algorithms in Python. Wiley, India.</li> <li>4. Marcello L. R. 2021. Advanced Algorithms and Data Structures, Manning, USA.</li> <li>5. Pai, G.A.V. 2008. Data Structures and Algorithms: Concepts, Techniques and Applications, Mc Graw Hill, USA.</li> </ol>

6. Weiss, M.A. 2011. Data Structures and Algorithm Analysis in Java. 3 <sup>rd</sup> Ed. Pearson, USA. 7. Weiss, M.A. 2013. Data Structures and Algorithm Analysis in C++. 4 <sup>th</sup> Ed. Pearson, USA.	6. Weiss, M.A. 2011. Data Structures and Algorithm Analysis in Java. 3 <sup>rd</sup> Ed. Pearson, USA. 7. Weiss, M.A. 2013. Data Structures and Algorithm Analysis in C++. 4 <sup>th</sup> Ed. Pearson, USA.
<b>IT-412</b> <b>Visual Programming</b> <b>3(2-1)</b>	<b>IT-412</b> <b>VISUAL PROGRAMMING</b> <b>3(2-1)</b>
<p><b>Objectives:</b>          To understand the basic application for visual programming, to understand and apply best practices for development of desktop applications. Constructing .NET desktop applications, accessing data with entity framework, Learning about modern Cross platform application development frameworks.</p> <p><b>Theory:</b>          Introduction: Microsoft C# and Visual Studio2019, Introduction to Windows Forms Application; Newer Trends in Desktop Application Development; Windows Presentation Forms; Developing .NET business applications; Object-oriented programming using C#; Designing classes with encapsulation, Applying inheritance, Employing polymorphism, Engaging the .NET framework, Common language runtime (CLR), Surveying the .NET framework class library (FCL), Programming .NET user interfaces, Designing User Interfaces using Extensible Application Markup Language (XAML); Constructing .NET desktop applications, Universal Windows Platform Applications, Xamarin as a cross platform development tool; Accessing data with entity framework, Connecting to databases with ADO.NET; Mapping objects to databases with ADO.NET; Entity framework; Integrating query into visual basic, Deploying applications, modern trends in desktop applications; Cross platform Development Technologies, business data analytics and visualization using desktop development tools</p> <p><b>Practical:</b>          Designing different programs using visual languages; Various Controls for Designing User Interface (UI); Navigation, Docking and Anchoring Features; Designing User Interface; Improving User Experience (UX); XAML for creating User Interfaces; Accessing database connectivity: ADO.Net to front end; Create, Read, Update and Delete (CRUD) operations on databases; Multiple Active Result Sets (MARS) Operations; Design Real Time Applications by using three-layer architectures and object oriented programming; Learning ML.NET</p> <p><b>Suggested Readings:</b>          1. Sharp, J. ,2018. Microsoft Visual C# Step by Step. Microsoft Press.          2. Clark, N. 2017. C#: Programming Basics for Absolute Beginners, Step-By-Step C#, Vol. 1, CreateSpace Independent Publishing Platform.</p>	<p><b>Course Learning Outcomes</b>          By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the basic application for visual programming, to understand and apply best practices for development of desktop applications.</li> <li>2. Construct .NET desktop applications, accessing data with entity framework, Learning about modern Cross platform application development frameworks.</li> <li>3. Express constants and arithmetic operations.</li> <li>4. Distinguish variable and data types.</li> <li>5. Code visual programs by using C#/Java work environment.</li> <li>6. Distinguish and compose events and methods.</li> </ol> <p><b>Theory</b>          Introduction: Microsoft C# and Visual Studio2019, Introduction to Windows Forms Application; Newer Trends in Desktop Application Development; Windows Presentation Forms; Developing .NET business applications; Object-oriented programming using C#; Designing classes with encapsulation, Applying inheritance, Employing polymorphism, Engaging the .NET framework, Common language runtime (CLR), Surveying the .NET framework class library (FCL), Programming .NET user interfaces, Designing User Interfaces using Extensible Application Markup Language (XAML); Constructing .NET desktop applications, Universal Windows Platform Applications, Xamarin as a cross platform development tool; Accessing data with entity framework, Connecting to databases with ADO.NET; Mapping objects to databases with ADO.NET; Entity framework; Integrating query into visual basic, Deploying applications, modern trends in desktop applications; Cross platform Development Technologies, business data analytics and visualization using desktop development tools</p> <p><b>Practical</b>          Designing different programs using visual languages; Various Controls for Designing User Interface (UI); Navigation, Docking and Anchoring Features; Designing User Interface; Improving User Experience (UX); XAML for creating User Interfaces; Accessing database connectivity: ADO.Net to front end; Create, Read, Update and Delete (CRUD) operations on databases; Multiple Active Result Sets (MARS) Operations; Design Real Time</p>

<ol style="list-style-type: none"> <li>Hermes, D., &amp; Mazloumi, N., 2019. Building Xamarin. Forms Apps Using XAML. In Building Xamarin. Forms Mobile Apps Using XAML (pp. 43-71). Apress, Berkeley, California, U.S.A.</li> <li>Mackey, A. 2010. Introducing .NET 4.0: With Visual Studio 2010. Apress, USA.</li> <li>Deitel, P., &amp; Deitel, H., 2016. Visual C# how to program. Pearson Press.</li> </ol>	<p>Applications by using three-layer architectures and object oriented programming; Learning ML.NET</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>Clark, N. 2017. C#: Programming Basics for Absolute Beginners, Step-By-Step C#, Vol. 1, CreateSpace Independent Publishing Platform.</li> <li>Sharp, J. ,2018. Microsoft Visual C# Step by Step. Microsoft Press.</li> <li>Hermes, D., and Mazloumi, N., 2019. Building Xamarin. Forms Apps Using XAML. In Building Xamarin. Forms Mobile Apps Using XAML (pp. 43-71). Apress, Berkeley, California, U.S.A.</li> <li>Mackey, A. 2010. Introducing .NET 4.0: With Visual Studio 2010. Apress, USA.</li> <li>Deitel, P., and Deitel, H., 2016. Visual C# how to program. Pearson Press.</li> </ol>
<b>IT-503</b> <b>Information Technology Infrastructure</b> <b>3(2-1)</b>	<b>IT-503</b> <b>INFORMATION TECHNOLOGY INFRASTRUCTURE</b> <b>3(2-1)</b>
<p><b>Objectives:</b> The course aims to provide students with a thorough understanding of the concept of an IT infrastructure, the components in an IT infrastructure, and the importance of information assurance and security in the selection, creation, integration and administration of an IT infrastructure.</p> <p><b>Theory:</b> Definition of IT Infrastructure; Non-functional Attributes; Availability Concepts; Sources of Unavailability, Availability Patterns. Performance; Security Concepts; Data centers; Servers: Availability, Performance and Security; Networking: Building Blocks, Availability, Performance and Security; Storage: Availability, Performance and Security; Virtualization: Availability, Performance and Security; Operating Systems: Building Blocks, Implementing Various OSs, OS availability, OS Performance and OS Security; End User Devices: Building Blocks, Device Availability, Performance and Security; IT Infrastructure Management; Service Delivery Processes; Service Support Processes; Ethics, Trends, organizational and technical issues related to IT infrastructure;</p> <p><b>Practical:</b> Install at least one modern operating system; server and network administration and management; Install network devices; Adjusting security services and their management.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>Sjaak, L., 2017. IT Infrastructure Architecture - Infrastructure Building Blocks and Concepts Third Edition.</li> <li>Gupta, P., 2009. IT Infrastructure and its Management. McGraw Hill Education Private Limited.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>Describe thorough understanding of the concept of an IT infrastructure, the components in an IT infrastructure, and the importance of information assurance and security in the selection, creation, integration and administration of an IT infrastructure.</li> <li>Explain through practical examples how protocols are used to enable communication between computing devices connected to each other.</li> <li>Express the role and structure of the Internet as an IT infrastructure component and design simple infrastructure solutions based on the use of the Internet.</li> <li>Define the role of IT control and service management frameworks in managing a large-scale organizational IT infrastructure solution.</li> </ol> <p><b>Theory</b> Definition of IT Infrastructure; Non-functional Attributes; Availability Concepts; Sources of Unavailability, Availability Patterns. Performance; Security Concepts; Data centers; Servers: Availability, Performance and Security; Networking: Building Blocks, Availability, Performance and Security; Storage: Availability, Performance and Security; Virtualization: Availability, Performance and Security; Operating Systems: Building Blocks, Implementing Various OSs, OS availability, OS Performance and OS Security; End User Devices: Building Blocks, Device Availability, Performance and Security; IT Infrastructure Management; Service Delivery Processes; Service Support Processes; Ethics, Trends, organizational and technical issues related to IT infrastructure;</p> <p><b>Practical</b></p>

<p>3. Hausman, K., 2010. IT Architecture for Dummies. 1st Edition.</p>	<p>Install at least one modern operating system; server and network administration and management; Install network devices; Adjusting security services and their management.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Sjaak, L., 2019. IT Infrastructure Architecture - Infrastructure Building Blocks and Concepts Third Edition.</li> <li>2. Gupta, P., 2009. IT Infrastructure and its Management. McGraw Hill Education Private Limited</li> <li>3. Hausman, K., 2010. IT Architecture for Dummies. 1st Edition.</li> </ol>
<p><b>IT-505                                      Machine Learning                                      3(2-1)</b></p>	<p><b>IT-505                                      MACHINE LEARNING                                      3(2-1)</b></p>
<p><b>Objectives:</b> The main objective of this course is to provide a broad survey of approaches and techniques in machine learning for developing a deeper understanding of programming skills that will help students to build intelligent, adaptive systems and help them pursue research in machine learning at later levels.</p> <p><b>Theory:</b> Basic concepts of Machine Learning: Types and Applications of Machine Learning, Artificial Intelligence vs Machine Learning; Essential Math for Machine Learning; Types of Machine Learning: Supervised learning, Unsupervised Learning, Deep Learning and Reinforcement Learning; Linear and Logistic regression; Decision Trees; Nearest Neighbor Algorithm; Support Vector Machines; Model selection and feature selection; Evaluating and debugging learning algorithms; Learning theory; Dimensionality Reduction using PCA (Principal components analysis); Linear Discriminant Analysis; K-means Clustering; Expectation Maximization algorithm. Factor analysis; Naïve Bayes; Ensemble Methods; Perceptron and Multilayer Perceptron; Artificial Neural Networks; Deep Learning; Loss functions; Activation Functions; Optimization Algorithms; Convolutional Neural Networks; Autoencoders; Generative adversarial networks; Recurrent Neural Networks; Reinforcement Learning; Quality learning.</p> <p><b>Practical:</b> How to handle data: Lists, Dictionaries in python, Loading datasets, Dataset preprocessing; Overview of NumPy and Pandas; Implementing Decision Trees; Regression; Nearest Neighbor Algorithm; Dimensionality reduction using Principal components analysis; Clustering using K-Means Algorithm; Support Vector Machine; Neural Networks Basics; Classification; Object Detection and Masking; Gradient and Gradient Descent; Loss Functions; Optimizers; Layers, Kernels and Filters; Pooling; Activation functions; Classifiers; Regularization techniques; Back Propagation; Convolutional Neural Networks;</p>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Demonstrate different approaches and techniques in machine learning.</li> <li>2. Apply machine learning problems corresponding to different applications.</li> <li>3. Explain a range of machine learning algorithms along with their strengths and weaknesses.</li> <li>4. Apply machine learning algorithms to solve problems of moderate complexity</li> </ol> <p><b>Theory</b> Basic concepts of Machine Learning: Types and Applications of Machine Learning, Artificial Intelligence vs Machine Learning; Essential Math for Machine Learning; Types of Machine Learning: Supervised learning, Unsupervised Learning, Deep Learning and Reinforcement Learning; Linear and Logistic regression; Decision Trees; Nearest Neighbor Algorithm; Support Vector Machines; Model selection and feature selection; Evaluating and debugging learning algorithms; Learning theory; Dimensionality Reduction using PCA (Principal components analysis); Linear Discriminant Analysis; K-means Clustering; Expectation Maximization algorithm. Factor analysis; Naïve Bayes; Ensemble Methods; Perceptron and Multilayer Perceptron; Artificial Neural Networks; Deep Learning; Loss functions; Activation Functions; Optimization Algorithms; Convolutional Neural Networks; Autoencoders; Generative adversarial networks; Recurrent Neural Networks; Reinforcement Learning; Quality learning.</p> <p><b>Practical</b> How to handle data: Lists, Dictionaries in python, Loading datasets, Dataset preprocessing; Overview of NumPy and Pandas; Implementing Decision Trees; Regression; Nearest Neighbor Algorithm; Dimensionality reduction using Principal components analysis; Clustering using K-Means Algorithm; Support Vector Machine; Neural Networks Basics; Classification; Object Detection and Masking; Gradient and Gradient Descent; Loss</p>



<p>Long Short Term Memory (LSTM) and Recurrent Neural Networks; Autoencoders; Transfer Learning; Reinforcement Learning.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Shukla, N., 2018. Machine learning with TensorFlow. Manning Publications Co.</li> <li>2. Kubat, M., 2017. An introduction to machine learning (Vol. 2). Springer International Publishing, Cham, Switzerland.</li> <li>3. Mueller, J. P., and Massaron, L., 2016. Machine learning for dummies. John Wiley &amp; Sons.</li> <li>4. Müller, A. C., and Guido, S., 2016. Introduction to machine learning with Python: a guide for data scientists. O'Reilly Media Inc.</li> </ol>	<p>Functions; Optimizers; Layers, Kernels and Filters; Pooling; Activation functions; Classifiers; Regularization techniques; Back Propagation; Convolutional Neural Networks; Long Short Term Memory (LSTM) and Recurrent Neural Networks; Autoencoders; Transfer Learning; Reinforcement Learning.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Shukla, N., 2019. Machine learning with TensorFlow. Manning Publications Co.</li> <li>2. Kubat, M., 2017. An introduction to machine learning (Vol. 2). Springer International Publishing, Cham, Switzerland.</li> <li>3. Mueller, J. P., and Massaron, L., 2016. Machine learning for dummies. John Wiley &amp; Sons.</li> <li>4. Müller, A. C., and Guido, S., 2016. Introduction to machine learning with Python: a guide for data scientists. O'Reilly Media Inc.</li> </ol>
<p><b>IT-507</b>                      <b>Database Administration and Management</b>                      <b>3(2-1)</b></p>	<p><b>IT-507</b>                      <b>DATABASE ADMINISTRATION AND MANAGEMENT</b>                      <b>3(2-1)</b></p>
<p><b>Objectives:</b> This course provides a foundation for database administration, exploring the fundamental models of database management systems. It covers the definition, development and management of databases for information systems.</p> <p><b>Theory:</b> Introduction to data administration and data management, data models such as object relational, object oriented, schema design, query languages and search specifications, including Structured Query Language (SQL), data structures, file organizations concepts, principles of data management systems, and hierarchical, network and relational data models, Transactional processing, Query processing and optimization, Database Programming, Integrity and security, Database Administration, Physical database design and tuning, Distributed database systems, Emerging research trends in database systems, storage Structures, User Security, Concurrency Control Mechanisms.</p> <p><b>Practical:</b> Database Auditing and Maintenance, Performance Management, Backup and Recovery techniques, Moving Data, Database Restart, Data Migration and its techniques.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Craig, S., 2012. Database Administration: The Complete Guide to DBA Practices and Procedures, 2nd Edition, Addison-Wesley Professional.</li> <li>2. Ahmad, O., 2019. Professional Azure SQL Database Administration: Equip yourself with the skills to manage and maintain data in the cloud, 2nd Edition</li> <li>3. Henry, F., 2011. Database System Concepts by, 6th edition, McGraw Hill.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the concepts and technical issues of database administration.</li> <li>2. Explain fundamental models of database management systems. It covers the definition, development, and management of databases for information systems.</li> <li>3. Explain understanding of internal functionality of DBMS.</li> <li>4. Perform database administration tasks like backup and recovery and performance tuning of databases.</li> </ol> <p><b>Theory</b> Introduction to data administration and data management, data models such as object relational, object oriented, schema design, query languages and search specifications, including Structured Query Language (SQL), data structures, file organizations concepts, principles of data management systems, and hierarchical, network and relational data models, Transactional processing, Query processing and optimization, Database Programming, Integrity and security, Database Administration, Physical database design and tuning, Distributed database systems, Emerging research trends in database systems, storage Structures, User Security, Concurrency Control Mechanisms.</p> <p><b>Practical</b> Database Auditing and Maintenance, Performance Management, Backup and Recovery techniques, Moving Data, Database Restart, Data Migration and its techniques.</p> <p><b>Suggested Readings</b></p>



			<div>1. Craig, S., 2012. Database Administration: The Complete Guide to DBA Practices and Procedures, 2nd Edition, Addison-Wesley Professional.</div> <div>2. Ahmad, O., 2019. Professional Azure SQL Database Administration: Equip yourself with the skills to manage and maintain data in the cloud, 2nd Edition</div> <div>3. Henry, F., 2011. Database System Concepts by, 6th edition, McGraw Hill.</div>
IT-509	Digital Signal Processing	3(2-1)	IT-509DIGITAL SIGNAL PROCESSING3(2-1)
<b>Theory:</b> Filtering, Linear Systems and Transforms; Filter Design and Implementation: IIR Filter; FIR Filter; Statistical Signal Processing: Correlation and Covariance; Nonparametric Methods; Parametric Methods; MUSIC and Eigenvector Analysis Methods; Kaiser Window; Chebyshev Window; Parametric Modeling; Resampling; Cepstrum Analysis; FFT-based Time-Frequency Analysis; Cross-Spectrogram of Complex Signals; Convolution and Correlation: Linear and Circular Convolution; Confidence Intervals; Residual Analysis; Cross-Correlations; Multirate Signal Processing: Downsampling; Upsampling; Spectral Analysis: Spectral Density Estimates; Bias and Variability in Periodogram; Cross-Spectrum and Magnitude-Squared Coherence; Amplitude Estimation; Harmonic Distortion; Linear Prediction: Prediction Polynomial; Formant Estimation; Partial Autocorrelation Sequence; Signal Generation; Signal Measurement: RMS Value; Slew Rate; Duty Cycle; Settling Time; Peak Amplitude; Distortion Measurement; Vibration Analysis; Signal Labeling.			<b>Course Learning Outcomes</b> By the end of this course students will be able to: <div><div>1. Identify the signals and systems</div><div>2. Apply the principles of discrete-time signal analysis to perform various signal operations</div><div>3. Apply the principles of z-transforms to finite difference equations.</div><div>4. Apply the principles of Fourier transform analysis to describe the frequency characteristics of discrete-time signals and systems</div><div>5. Apply the principles of signal analysis to filtering</div><div>6. Use computer programming tools to process and visualize signals</div></div>
<b>Practical:</b> Signal Generation and Visualization; Signal Smoothing; Resampling; Align Signals with Different Start Times; Signal Reconstruction; Designing a Filter in fdesign (MATLAB); Designing a Filter in Filter Builder GUI; Comparison of Filters; Filter Designer and Analysis App; SPTool: Filter Visualization; Spectrum Viewer; Filtering and Analysis of Noise; Importing and Exporting Filters; Code Generation; Signal Analyzer App.			<b>Theory</b> Filtering, Linear Systems and Transforms; Filter Design and Implementation: IIR Filter; FIR Filter; Statistical Signal Processing: Correlation and Covariance; Nonparametric Methods; Parametric Methods; MUSIC and Eigenvector Analysis Methods; Kaiser Window; Chebyshev Window; Parametric Modeling; Resampling; Cepstrum Analysis; FFT-based Time-Frequency Analysis; Cross-Spectrogram of Complex Signals; Convolution and Correlation: Linear and Circular Convolution; Confidence Intervals; Residual Analysis; Cross-Correlations; Multirate Signal Processing: Downsampling; Upsampling; Spectral Analysis: Spectral Density Estimates; Bias and Variability in Periodogram; Cross-Spectrum and Magnitude-Squared Coherence; Amplitude Estimation; Harmonic Distortion; Linear Prediction: Prediction Polynomial; Formant Estimation; Partial Autocorrelation Sequence; Signal Generation; Signal Measurement: RMS Value; Slew Rate; Duty Cycle; Settling Time; Peak Amplitude; Distortion Measurement; Vibration Analysis; Signal Labeling.
<b>Suggested Readings:</b> <div><div>1. Downey, A. B. 2016. Think DSP: Digital Signal Processing in Python. O'Reilly Media, Inc. USA.</div><div>2. Lyons, R. G., and D. L. Fugal. 2014. The Essential Guide to Digital Signal Processing. Prentice Hall. USA.</div><div>3. Newbold, R. 2012. Practical Applications in Digital Signal Processing. Prentice Hall. USA.</div><div>4. Signal Processing Toolbox TM User’s Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</div><div>5. Vetterli, M., J. Kovačević, V. K. Goyal. 2014. Foundations of Signal Processing. Cambridge University Press. UK.</div></div>			<b>Practical</b> Signal Generation and Visualization; Signal Smoothing; Resampling; Align Signals with Different Start Times; Signal Reconstruction; Designing a Filter in design (MATLAB); Designing a Filter in Filter Builder GUI; Comparison of Filters; Filter Designer and Analysis App; SPTool: Filter Visualization; Spectrum Viewer; Filtering and Analysis of Noise; Importing and Exporting Filters; Code Generation; Signal Analyzer App.
			<b>Suggested Readings</b>

			<div><div><div>1. Downey, A. B. 2016. Think DSP: Digital Signal Processing in Python. O'Reilly Media, Inc. USA.</div><div>2. Lyons, R. G., and D. L. Fugal. 2014. The Essential Guide to Digital Signal Processing. Prentice Hall. USA.</div><div>3. Newbold, R. 2012. Practical Applications in Digital Signal Processing. Prentice Hall. USA.</div><div>4. Signal Processing Toolbox TM User’s Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</div><div>5. Vetterli, M., J. Kovačević, V. K. Goyal. 2014. Foundations of Signal Processing. Cambridge University Press. UK.</div></div></div>		
IT-511	Artificial Intelligence	3(2-1)	IT-511ARTIFICIAL INTELLIGENCE3(2-1)		
<div>Theory: Introduction; Turing Machine; Knowledge Representation; Search Methodologies: Depth-First; Breadth-First; Properties of Search Methods; Heuristics; Hill Climbing; Best-First; Beam Search; Greedy Search; Simulated Annealing; Parallel Search; Game Playing; Propositional and Predicate Logic; Inference and Resolution for Problem Solving; Rules and Expert Systems; Machine Learning: Decision Tree; Supervised Learning; Unsupervised Learning; Re-inforcement Learning; Neural Networks: Perceptron; Back-propagation; Multilayer Networks; Recurrent Networks; Probabilistic Reasoning; Bayesian Belief Networks; Fuzzy Reasoning: Fuzzy Sets; Fuzzy Logic; Fuzzy Inference; Genetic Algorithms; Intelligent Agents; Natural Language Processing Basics; Machine Vision: Image Processing; Texture Analysis; Motion Estimation; Face Recognition; Robotics.</div> <div>Practical: Neural Networks: Designing, Developing and Training Neural Networks for Function Approximation, Pattern Recognition, Clustering and Time Series Modeling; Fuzzy Logic: Building Fuzzy Systems using Fuzzy Logic Designer, Command Line and Custom Functions; Adaptive Neuro-Fuzzy Modeling; Fuzzy Clustering; Deploying Fuzzy Inference Systems; Implementing Mamdani-Type and Sugeno-Type Fuzzy Inference Systems; Logic Image Processing.</div> <div>Suggested Readings:<div><div>1. Daugherty, R. R., and H. J. Wilson. 2018. Human + Machine: Reimagining Work in the Age of AI. Harvard Business Review Press. USA.</div><div>2. Marr, B., and M. Ward. 2019. Artificial Intelligence in Practice. John Wiley &amp; Sons. USA.</div><div>3. Kaplan, J., 2016. 2016. Artificial Intelligence: What Everyone Needs to Know. Oxford University Press. UK.</div></div></div>			<div>Course Learning Outcomes By the end of this course students will be able to:<div><div>1. Explain the applications of AI and agent-based approach to AI.</div><div>2. Demonstrate first-order predicate calculus, logical reasoning and problem-solving using Prolog language.</div><div>3. Apply various techniques and algorithms of AI used in general problem solving, optimization problems, constraint satisfaction problems, and game programming.</div></div></div> <div>Theory Introduction; Turing Machine; Knowledge Representation; Search Methodologies: Depth-First; Breadth-First; Properties of Search Methods; Heuristics; Hill Climbing; Best-First; Beam Search; Greedy Search; Simulated Annealing; Parallel Search; Game Playing; Propositional and Predicate Logic; Inference and Resolution for Problem Solving; Rules and Expert Systems; Machine Learning: Decision Tree; Supervised Learning; Unsupervised Learning; Re-inforcement Learning; Neural Networks: Perceptron; Back-propagation; Multilayer Networks; Recurrent Networks; Probabilistic Reasoning; Bayesian Belief Networks; Fuzzy Reasoning: Fuzzy Sets; Fuzzy Logic; Fuzzy Inference; Genetic Algorithms; Intelligent Agents; Natural Language Processing Basics; Machine Vision: Image Processing; Texture Analysis; Motion Estimation; Face Recognition; Robotics.</div> <div>Practical Neural Networks: Designing, Developing and Training Neural Networks for Function Approximation, Pattern Recognition, Clustering and Time Series Modeling; Fuzzy Logic: Building Fuzzy Systems using Fuzzy Logic Designer, Command Line and Custom Functions; Adaptive Neuro-Fuzzy Modeling; Fuzzy Clustering; Deploying Fuzzy Inference Systems; Implementing Mamdani-Type and Sugeno-Type Fuzzy Inference Systems; Logic Image Processing.</div>		

<ol style="list-style-type: none"> <li>Govers, F. X., 2018. Artificial Intelligence for Robotics. Packt Publishing. UK.</li> <li>Neural Network Toolbox™ User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> <li>Fuzzy Logic Toolbox™ User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> </ol>	<p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>Daugherty, R. R., and H. J. Wilson. 2018. Human + Machine: Reimagining Work in the Age of AI. Harvard Business Review Press. USA.</li> <li>Fuzzy Logic Toolbox™ User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> <li>Govers, F. X., 2018. Artificial Intelligence for Robotics. Packt Publishing. UK.</li> <li>Kaplan, J., 2016. 2016. Artificial Intelligence: What Everyone Needs to Know. Oxford University Press. UK.</li> <li>Marr, B., and M. Ward. 2019. Artificial Intelligence in Practice. John Wiley &amp; Sons. USA.</li> <li>Neural Network Toolbox™ User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> <li></li> </ol>
<b>IT-513</b> <b>Web Programming</b> <b>3(2-1)</b>	<b>IT-513</b> <b>WEB PROGRAMMING</b> <b>3(2-1)</b>
<p><b>Objectives:</b> In this course, students will learn the basics of designing and developing a web site. We'll take a look at design issues specific to web-based presentations, learn web page layout, effective navigation and delve into the design process.</p> <p><b>Theory:</b> Introduction to E-commerce, TCP/IP, HTTP, Overview of 3-Tier architecture, Web based applications; Architecture, and developing front end applications tools, HTML, Hierarchical structure: elements (tags) and text, basic XHTML document structure, Paragraphs, Whitespace and line breaks in XHTML, Headings, Phrase mark-up in XHTML, Hypertext links, Embedded images, Lists, Preformatted text, Special characters in XHTML, XHTML and Semantic markup, Content design; DHTML; Introduction to Dot Net/Php/C#; Scripting (Java Script, VB-Script), Java Applets, Active-X.</p> <p><b>Practical:</b> HTML tags (Head, body, title A background, color, border, table, picture etc.), framing tags, creating static pages using tags, DHTML, CSS, creating dynamic pages using ASP. NET/PHP. Designing multiple statics/dynamic pages, client/server architecture, applet, servlet.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>Esposito, D., 2018. Programming ASP. NET Core, Programming ASP. NET Core. Microsoft Press.</li> <li>Duckett, J., 2014. HTML and CSS: Design and Build Websites. McGraw Hill CA, USA.</li> </ol>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>Implement graphic design principles that relate to web design and learn how to implement theories into practice.</li> <li>Apply skills in analyzing the usability of a web site.</li> <li>Describe how to plan and conduct user research related to web usability.</li> <li>Apply techniques of responsive web design, including media queries.</li> <li>Demonstrate basic programming skills using JavaScript and jQuery.</li> </ol> <p><b>Theory</b> Introduction to E-commerce, TCP/IP, HTTP, Overview of 3-Tier architecture, Web based applications; Architecture, and developing front end applications tools, HTML, Hierarchical structure: elements (tags) and text, basic XHTML document structure, Paragraphs, Whitespace and line breaks in XHTML, Headings, Phrase mark-up in XHTML, Hypertext links, Embedded images, Lists, Preformatted text, Special characters in XHTML, XHTML and Semantic markup, Content design; DHTML; Introduction to Dot Net/Php/C#; Scripting (Java Script, VB-Script), Java Applets, Active-X.</p> <p><b>Practical</b> HTML tags (Head, body, title A background, color, border, table, picture etc.), framing tags, creating static pages using tags, DHTML, CSS, creating dynamic pages using ASP. NET/PHP. Designing multiple statics/dynamic pages, client/server architecture, applet, servlet.</p> <p><b>Suggested Readings</b></p>

<ol style="list-style-type: none"> <li>Deitel, H., 2016. Java How to Program. Prentice Hall International, USA.</li> <li>Naughton, P., 2012. The Complete Reference Java-2, McGraw Hill CA, USA.</li> </ol>	<ol style="list-style-type: none"> <li>Duckett, J., 2014. HTML and CSS: Design and Build Websites. McGraw Hill CA, USA.</li> <li>Deitel, H., 2016. Java How to Program. Prentice Hall International, USA.</li> <li>Esposito, D., 2019. Programming ASP. NET Core, Programming ASP. NET Core. Microsoft Press.</li> <li>Naughton, P., 2012. The Complete Reference Java-2, McGraw Hill CA, USA.</li> </ol>
IT-502                      Data Encryption and Security                      3(2-1)	IT-502                      DATA ENCRYPTION AND SECURITY                      3(2-1)
<p><b>Objectives:</b> The course aims to equip students with the ability to understand computer and network security, perform encryption and decryption using different algorithms, encrypt messages using public-key cryptosystems, create digital signatures using available algorithms, perform email security.</p> <p><b>Theory:</b> Computer Security Concepts; The OSI Security Architecture; Security Attacks; Security Services; Security Mechanisms; A Model for Network Security; Symmetric Cipher Model; Substitution Techniques; Transposition Techniques; Steganography; Rotor Machines; Traditional Block Cipher Structure; The Data Encryption Standard; Block Cipher Design Principles; Basic Concepts in Number Theory and Finite Fields: Divisibility and the Division Algorithm, The Euclidean Algorithm, Modular Arithmetic, Groups, Rings, Fields, Finite Fields of the Form <math>GF(p)</math>, Polynomial Arithmetic, Finite Fields of the Form <math>GF(2^n)</math>, Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem; Finite Field Arithmetic; AES Algorithm: Structure, Transformation, Expansion, Implementation; Multiple Encryption and Triple DES; Electronic Code book; Cipher Block Chaining Mode; Cipher Feedback Mode; Output Feedback Mode; Counter Mode; Principles of Pseudorandom Number Generation; Pseudorandom Number Generators; Pseudorandom Number Generation Using a Block Cipher; Stream Ciphers; True Random Number Generators; Principles of Public-Key Cryptosystems; The RSA Algorithm; Diffie-Hellman Key Exchange; Elgamal Cryptographic System; Elliptic Curve Arithmetic and Cryptography; Pseudorandom Number Generation Based on an Asymmetric Cipher; Digital Signatures; Elgamal Digital Signature Scheme; Schnorr Digital Signature Scheme; NIST Digital Signature Algorithm; Elliptic Curve Digital Signature Algorithm; RSA-PSS Digital Signature Algorithm; Electronic Mail Security;</p> <p><b>Practical:</b> Implementation of Cryptography algorithms: Caesar Cipher, Monoalphabetic Cipher, DES Encryption and Decryption, Double and Triple DES, Storage Encryption, Pseudorandom Number Generators.</p>	<p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>Apply and implement encryption in practice.</li> <li>Implement fundamental principles and theories underlying encryption algorithms, including the mathematical foundations of encryption.</li> <li>Demonstrate skills to apply cryptography to solving data security problems</li> <li>Implement encryption algorithms and protocols work and how to use them.</li> <li>Apply a broad view of security with practical applications of encryption to data security.</li> </ol> <p><b>Theory</b> Computer Security Concepts; The OSI Security Architecture; Security Attacks; Security Services; Security Mechanisms; A Model for Network Security; Symmetric Cipher Model; Substitution Techniques; Transposition Techniques; Steganography; Rotor Machines; Traditional Block Cipher Structure; The Data Encryption Standard; Block Cipher Design Principles; Basic Concepts in Number Theory and Finite Fields: divisibility and the division algorithm, the Euclidean algorithm, modular arithmetic, groups, rings, fields, finite fields of the form <math>gf(p)</math>, polynomial arithmetic, finite fields of the form <math>GF(2^n)</math>, prime numbers, Fermat's and Euler's theorems, testing for primality, the Chinese remainder theorem; Finite Field Arithmetic; AES Algorithm: structure, transformation, expansion, implementation; Multiple Encryption and Triple DES; Electronic Code book; Cipher Block Chaining Mode; Cipher Feedback Mode; Output Feedback Mode; Counter Mode; Principles of Pseudorandom Number Generation; Pseudorandom Number Generators; Pseudorandom Number Generation Using a Block Cipher; Stream Ciphers; True Random Number Generators; Principles of Public-Key Cryptosystems; The RSA Algorithm; Diffie-Hellman Key Exchange; Elgamal Cryptographic System; Elliptic Curve Arithmetic and Cryptography; Pseudorandom Number Generation Based on an Asymmetric Cipher; Digital Signatures; Elgamal Digital Signature Scheme; Schnorr Digital Signature Scheme; NIST Digital Signature Algorithm; Elliptic Curve Digital Signature Algorithm; RSA-PSS Digital Signature Algorithm; Electronic Mail Security;</p>

<p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Aumasson, J. P. 2017. Serious Cryptography: A Practical Introduction to Modern Encryption. USA.</li> <li>2. Bishop, M. 2018. Computer Security: Art and Science. 2nd Ed. Addison-Wesley, USA.</li> <li>3. Kaufman, R. and M, Speciner. 2002. Network Security: Private Communication in a Public World. 2nd Ed. Prentice Hall, USA.</li> <li>4. Paar, C. and J. Pelzl. 2009. Understanding cryptography: A Textbook for Students and Practitioners. Springer, USA.</li> <li>5. Richard, A.M. 2002. An Introduction to Cryptography. 2nd Ed. Chapman and Hall, CRC Press, USA.</li> <li>6. Stallings, W. 2016. Cryptography and Network Security. 7th Ed. Pearson, USA.</li> </ol>	<p><b>Practical</b></p> <p>Implementation of Cryptography algorithms: Caesar Cipher, Monoalphabetic Cipher, DES Encryption and Decryption, Double and Triple DES, Storage Encryption, Pseudorandom Number Generators.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Aumasson, J. P. 2017. Serious Cryptography: A Practical Introduction to Modern Encryption. USA.</li> <li>2. Bishop, M. 2019. Computer Security: Art and Science. 2nd Ed. Addison-Wesley, USA.</li> <li>3. Paar, C. and J. Pelzl. 2009. Understanding cryptography: A Textbook for Students and Practitioners. Springer, USA.</li> <li>4. Richard, A.M. 2002. An Introduction to Cryptography. 2nd Ed. Chapman and Hall, CRC Press, USA.</li> <li>5. Stallings, W. 2016. Cryptography and Network Security. 7th Ed. Pearson, USA.</li> </ol>
<p><b>IT-504</b>                      <b>Digital Image Processing</b>                      <b>3(2-1)</b></p>	<p><b>IT-504</b>                      <b>DIGITAL IMAGE PROCESSING</b>                      <b>3(2-1)</b></p>
<p><b>Theory:</b></p> <p>Introduction; Image Formation; Image Coordinate Systems; Image Types; Image Conversion; Image Arithmetic Functions; Multispectral Images; Reading and Writing Image Data; Image Display; Geometric Transformations; Image Registration; Image Filtering; Fourier Transform; Spatial and Frequency Domain Processing; Discrete Cosine Transform; Hough Transform; Radon Transform; Edge Detection; Corner Detection; Line Detection; Morphological Dilation and Erosion; Structuring Elements; Pixel Connectivity; Region Detection; Image Histogram; Texture Analysis; Gamma Correction; Contrast Enhancement; Histogram Equalization; Adaptive Histogram Equalization; Image Smoothing; Noise Removal; ROI-based Processing; Image Segmentation; Gabor Filters; Texture Filters; Color-based Segmentation; Image Deblurring; Color Spaces; Neighborhood and Block Operations; Deep Learning; Denoising Images using Neural Networks; Semantic Segmentation using Deep Learning; High-performance Computing Paradigm for Image Processing and Computer Vision.</p> <p><b>Practical:</b></p> <p>Exploring Image using Image Viewer App (MATLAB); Reading, Writing and Querying Images; Basic Display of Images, Accessing Pixel Values; Histogram Calculation; Filtering: Mean; Median; Rank; Gaussian; Edge Detection; Corner Detection; Segmentation; FFT; DFT; Region Analysis; Texture Analysis; Color Space Conversions; Feature Detection and Extraction; GPU Computing.</p> <p><b>Suggested Readings:</b></p>	<p><b>Course Learning Outcomes</b></p> <p>By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Apply the concepts of image processing and basic analytical methods to be used in image processing.</li> <li>2. Implement image enhancement and restoration techniques,</li> <li>3. Demonstrate different image compression techniques.</li> <li>4. Explain segmentation and morphological processing techniques.</li> </ol> <p><b>Theory</b></p> <p>Introduction; Image Formation; Image Coordinate Systems; Image Types; Image Conversion; Image Arithmetic Functions; Multispectral Images; Reading and Writing Image Data; Image Display; Geometric Transformations; Image Registration; Image Filtering; Fourier Transform; Spatial and Frequency Domain Processing; Discrete Cosine Transform; Hough Transform; Radon Transform; Edge Detection; Corner Detection; Line Detection; Morphological Dilation and Erosion; Structuring Elements; Pixel Connectivity; Region Detection; Image Histogram; Texture Analysis; Gamma Correction; Contrast Enhancement; Histogram Equalization; Adaptive Histogram Equalization; Image Smoothing; Noise Removal; ROI-based Processing; Image Segmentation; Gabor Filters; Texture Filters; Color-based Segmentation; Image Deblurring; Color Spaces; Neighborhood and Block Operations; Deep Learning; Denoising Images using Neural Networks; Semantic Segmentation using Deep Learning; High-performance Computing Paradigm for Image Processing and Computer Vision.</p>

<ol style="list-style-type: none"> <li>1. Gonzalez, R. C., and R. E. Woods. 2008. Digital Image Processing. Prentice Hall. USA.</li> <li>2. Image Processing Toolbox TM User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> <li>3. Joshi, M. A. 2018. Digital Image Processing: An Algorithmic Approach. PHI Learning (pvt.). India.</li> <li>4. Sundararajan, D. 2017. Digital Image Processing: A Signal Processing and Algorithmic Approach. Springer Nature Singapore.</li> <li>5. Solomon, C., and T. Breckon. 2011. Fundamentals of Digital Image Processing: A Practical Approach with Examples in MATLAB. John Wiley &amp; Sons. UK.</li> </ol>	<p><b>Practical</b> Exploring Image using Image Viewer App (MATLAB); Reading, Writing and Querying Images; Basic Display of Images, Accessing Pixel Values; Histogram Calculation; Filtering: Mean; Median; Rank; Gaussian; Edge Detection; Corner Detection; Segmentation; FFT; DFT; Region Analysis; Texture Analysis; Color Space Conversions; Feature Detection and Extraction; GPU Computing.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Gonzalez, R. C., and R. E. Woods. 2008. Digital Image Processing. Prentice Hall. USA.</li> <li>2. Image Processing Toolbox TM User's Guide. 2019. The MathWorks, Inc. 1 Apple Hill Drive, Natick, MA, USA.</li> <li>3. Joshi, M. A. 2018. Digital Image Processing: An Algorithmic Approach. PHI Learning (pvt.). India.</li> <li>4. Sundararajan, D. 2017. Digital Image Processing: A Signal Processing and Algorithmic Approach. Springer Nature Singapore.</li> <li>5. Solomon, C., and T. Breckon. 2011. Fundamentals of Digital Image Processing: A Practical Approach with Examples in MATLAB. John Wiley &amp; Sons. UK.</li> </ol>
<p><b>IT-506</b> <b>Big Data Analytics</b> <b>3(2-1)</b></p> <p><b>Theory:</b> Emergence of data: evolution of sciences; data science and its process; evolution of database technology; data warehouse use of OLAP; Big data: definition and taxonomy; Characteristics of big data; big data and data mining; characteristics of structured and unstructured data; Data Analytics types: predictive analytics, descriptive analytics and prescriptive analytics; reasoning methods: deductive reasoning and inductive reasoning; business intelligence; Expert system and its architecture: knowledge base, inference engine: forward chaining and backward chaining; concept and use of Predicate logic; Business intelligence; Introduction to Hadoop: architecture, difference between RDBMS and Hadoop; Graph Theory: node, link, directed graph, undirected graph, weakly connected components, strongly connected components; social network analysis: concept and centrality measures; Machine learning: clustering, association, classification, naïve baise and others.</p> <p><b>Practical:</b> Understanding of big data analytical tool; working with different data sets: explore and normalization of datasets; Use modern programming language for understanding and implementation of algorithms: k-means, association, classification, naïve baise and others; centrality measures: degree, closeness and betweenness; Case studies;</p>	<p><b>IT-506</b> <b>BIG DATA ANALYTICS</b> <b>3(2-1)</b></p> <p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the basic technologies that forms the foundations of Big Data</li> <li>2. Implement programming aspects of cloud computing with a view to rapid prototyping of complex applications.</li> <li>3. Apply specialized aspects of big data including big data application, and big data analytics.</li> <li>4. Demonstrate different case studies on the current research and applications of the Hadoop and big data in industry</li> </ol> <p><b>Theory</b> Emergence of data: evolution of sciences; data science and its process; evolution of database technology; data warehouse use of OLAP; Big data: definition and taxonomy; Characteristics of big data; big data and data mining; characteristics of structured and unstructured data; Data Analytics types: predictive analytics, descriptive analytics and prescriptive analytics; reasoning methods: deductive reasoning and inductive reasoning; business intelligence; Expert system and its architecture: knowledge base, inference engine: forward chaining and backward chaining; concept and use of Predicate logic; Business intelligence; Introduction to Hadoop: architecture, difference between RDBMS and Hadoop;</p>



<p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Aggarwal C. C. 2015. Data Mining: The Textbook, IBM T.J. Watson Research Center, New York, USA.</li> <li>2. Dean J. 2014. Big Data, Data Mining, and Machine Learning, Value Creation for Business Leaders and Practitioners, Published by John Wiley and Sons, Inc., New Jersey, Canada.</li> <li>3. Kaya M., J. Kawash, S. Khoury and M-Y Day, 2018. Social Network Based Big Data Analysis and Applications, 1<sup>st</sup> Ed. Springer International Publishing, Switzerland.</li> <li>4. Reynolds, V. 2016. Big Data for Beginners: Understanding Smart Big Data, Data Mining &amp; Data Analytics for Improved Business Performance, Life Decisions &amp; More. Kindle ed. Scotts Valley, California, United States.</li> <li>5. Reynolds, V. 2016. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. 1st Ed. Wiley EMC education series, New Jersey, USA.</li> <li>6. Schonberger V. M. and K. Cukier, 2017. Big Data: The Essential Guide to Work, Life and Learning in the Age of Insight, kindle Ed. John Murray, New York, USA.</li> </ol>	<p>Graph Theory: node, link, directed graph, undirected graph, weakly connected components, strongly connected components; social network analysis: concept and centrality measures; Machine learning: clustering, association, classification, naïve Bayes and others.</p> <p><b>Practical</b></p> <p>Understanding of big data analytical tool; working with different data sets: explore and normalization of datasets; Use modern programming language for understanding and implementation of algorithms: k-means, association, classification, naïve Bayes and others; centrality measures: degree, closeness and betweenness; Case studies;</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Aggarwal C. C. 2015. Data Mining: The Textbook, IBM T.J. Watson Research Center, New York, USA.</li> <li>2. Dean J. 2014. Big Data, Data Mining, and Machine Learning, Value Creation for Business Leaders and Practitioners, Published by John Wiley and Sons, Inc., New Jersey, Canada.</li> <li>3. Kaya M., J. Kawash, S. Khoury and M-Y Day, 2019. Social Network Based Big Data Analysis and Applications, 1<sup>st</sup> Ed. Springer International Publishing, Switzerland.</li> <li>4. Reynolds, V. 2016. Big Data for Beginners: Understanding Smart Big Data, Data Mining &amp; Data Analytics for Improved Business Performance, Life Decisions &amp; More. Kindle ed. Scotts Valley, California, United States.</li> <li>5. Reynolds, V. 2016. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. 1st Ed. Wiley EMC education series, New Jersey, USA.</li> <li>6. Schonberger V. M. and K. Cukier, 2017. Big Data: The Essential Guide to Work, Life and Learning in the Age of Insight, kindle Ed. John Murray, New York, USA.</li> </ol>
<p><b>IT-508</b> <b>Cloud Computing</b> <b>3(2-1)</b></p> <p><b>Theory</b></p> <p>Introduction to Cloud Computing: Components of a computing cloud, differentiating types of clouds: public, private, hybrid, Delivering services from the cloud. Adopting the Cloud: Key drivers of cloud computing solutions, evaluating barriers to cloud computing. Exploiting Software as a Service (SaaS): Characterizing SaaS, comparing service scenarios, Inspecting SaaS technologies. Delivering Platform as a Service (PaaS): Exploring the technical foundation for PaaS, building services with solution stacks, Managing cloud storage, Employing support services. Deploying Infrastructure as a Service (IaaS): Enabling technologies, Accessing IaaS. Building a Business Case: Calculating the financial</p>	<p><b>IT-508</b> <b>CLOUD COMPUTING</b> <b>3(2-1)</b></p> <p><b>Course Learning Outcomes</b></p> <p>By the end of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Describe the necessary theoretical background for computing and storage clouds environments.</li> <li>2. Explain the methodologies and technologies for the development of applications that will be deployed and offered through cloud computing environments.</li> <li>3. Realize cloud infrastructures by using IaaS software, while also developing cloud applications by utilizing PaaS software.</li> </ol> <p><b>Theory</b></p>

<p>implications, Preserving business continuity. Migrating to the Cloud: Technical considerations. Cloud Security: Cloud Security Attributes, Security as a Service from the Cloud, Cloud and Security Risks. Cloud Architectural Models: Design for Failure (DFF), Traditional.</p> <p><b>Practical</b> Building Enterprise Cloud Computing Environment using a Network of Computers, Platform for building Clouds and their Innovative Application, Cloud Computing Platforms, Guidelines for setting up of CLOUDS LAB; scripts of HDFS: Hadoop; MapReduce, and its Java API. Hadoop Distributed File System (HDFS) features.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Bahga, A. and V. Madiseti. 2019. Cloud Computing Solutions Architect: A Hands-On Approach: A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond. VPT, Georgia, USA.</li> <li>2. Kavis, M.J. 2016. Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS). Kindle Edition, John Wiley &amp; Sons, Inc., Hoboken, New Jersey, USA.</li> <li>3. Marks, E.A. and B. Lozano. 2018. Executive's guide to cloud computing. John Wiley &amp; Sons, Inc., Hoboken, New Jersey</li> <li>4. Rafaels, R. 2018. Cloud Computing 2nd Edition. Create Space Independent Publishing Platform. Scotts Valley, California, USA.</li> <li>5. Wang, L., R. Ranjan, J. Chen and B. Benatallah. 2017. Cloud Computing: Methodology, Systems, and Applications. Taylors and Frances Group, Milton Park, Didcot, UK.</li> </ol>	<p>Introduction to Cloud Computing: components of a computing cloud, differentiating types of clouds: public, private, hybrid, delivering services from the cloud; Adopting the Cloud: key drivers of cloud computing solutions, evaluating barriers to cloud computing; Exploiting Software as a Service (SaaS): Characterizing SaaS, comparing service scenarios, Inspecting SaaS technologies; Delivering Platform as a Service (PaaS): exploring the technical foundation for PaaS, Building services with solution stacks, Managing cloud storage, Employing support services. Deploying Infrastructure as a Service (IaaS): Enabling technologies, Accessing IaaS. Building a Business Case: Calculating the financial implications, Preserving business continuity. Migrating to the Cloud: Technical considerations. Cloud Security: Cloud Security Attributes, Security as a Service from the Cloud, Cloud and Security Risks. Cloud Architectural Models: Design for Failure (DFF), Traditional.</p> <p><b>Practical</b> Building Enterprise Cloud Computing Environment using a Network of Computers, Platform for building Clouds and their Innovative Application, Cloud Computing Platforms, Guidelines for setting up of CLOUDS LAB; scripts of HDFS: Hadoop; MapReduce, and its Java API. Hadoop Distributed File System (HDFS) features.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Bahga, A. and V. Madiseti. 2019. Cloud Computing Solutions Architect: A Hands-On Approach: A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond. VPT, Georgia, USA.</li> <li>2. Kavis, M.J. 2016. Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS). Kindle Edition, John Wiley &amp; Sons, Inc., Hoboken, New Jersey, USA.</li> <li>3. Marks, E.A. and B. Lozano. 2018. Executive's guide to cloud computing. John Wiley &amp; Sons, Inc., Hoboken, New Jersey</li> <li>4. Rafaels, R. 2018. Cloud Computing 2nd Edition. Create Space Independent Publishing Platform. Scotts Valley, California, USA.</li> <li>5. Wang, L., R. Ranjan, J. Chen and B. Benatallah. 2017. Cloud Computing: Methodology, Systems, and Applications. Taylors and Frances Group, Milton Park, Didcot, UK.</li> </ol>
<p><b>IT-510</b> <b>Internet of Things</b> <b>3(2-1)</b></p> <p><b>Objectives:</b></p>	<p><b>IT-510</b> <b>INTERNET OF THINGS</b> <b>3(2-1)</b></p> <p><b>Course Learning Outcomes</b> By the end of this course students will be able to:</p>



<p>The course aims to provide students with a thorough understanding of the concept interconnection and integration of the physical world and the cyber space. They are also able to design &amp; develop IOT Devices.</p> <p><b>Theory:</b> Introduction of Internet of Things (IoT); Need of IoT Security; Requirement and Basic Properties; Main Challenges: Confidentiality, Integrity, Availability and Non-Repudiation; IoT Architectures (Device, Cloud, Gateway, Backend, Applications); Security Classification &amp; Access Control; Data classification: Public and Private; Privacy issues in IoT; IoT Authentication and Authorization; IoT Data Integrity; Web Based Attacks and Implementation in IoT; Denial of Service; Sniffing, Phishing, DNS Hijacking, Pharming and Defacement; Cryptology Cipher; Symmetric Key Algorithms (AES and DES); Asymmetric Key Algorithm (RSA); Attacks: Dictionary and Brute Force, Lookup Tables, Reverse Lookup Tables and Rainbow Tables; Attack Surface in IoT and Threat Assessment; Embedded Device: UART, SPI, I2C, JTAG; Software and Cloud components: Firmware of the device; Web Application Dashboard; Mobile Application used to Control; Configure and Monitor the devices; Radio Communication: WIFI, BLE, Cellular, ZigBee*, ZWave* and 6LoWPAN; IoT Protocol inbuilt Security Features on Transport Layer and Application Layer; Security Management; Identity and Access Management; Key Management and Model Discussions: Smart Home, Smart Agriculture, Smart Retail Supply, Smart Healthcare, Smart Grid, and Smart Cities;</p> <p><b>Practical:</b> Developing Internet of Things &amp; Logical Design using Python: Introduction, IoT Design Methodology, Installing Python, Python Data Types &amp; Data Structures, Control Flow, Functions, Modules, Packages, File Handling, Date/Time Operations, Classes and Python Packages; IoT Physical Devices &amp; Endpoints: What is an IoT Device, Exemplary Device, Board, Linux on Raspberry Pi, Interfaces, and Programming and IoT Devices.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Hanes, D., G. Salgueiro, R. Barton, J. Henry and P. Grossetete. 2017. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things. Cisco Press. Indianapolis, Indiana, USA.</li> <li>2. Khan, J.Y. and M.R. Yuce. 2019. Internet of Things (iot). Pan Stanford Publishing Pte Limited, Central Tower, Temasek Avenue, Singapore.</li> <li>3. Kranz, M. 2016. Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry. John Wiley &amp; Sons, Inc., Hoboken, New Jersey, USA.</li> </ol>	<ol style="list-style-type: none"> <li>1. Demonstrate the understanding of the concept interconnection and integration of the physical world and the cyber space.</li> <li>2. Implement Internet of Things (IoT) is the network of physical objects in which microprocessor and wireless radios are embedded to intelligently serve people in a collaborative manner.</li> <li>3. Apply the concept and capabilities of smart thing/object/device identification, and physical principles of sensing.</li> <li>4. Explain IoT enabling technologies, architectures, and standards</li> </ol> <p><b>Theory</b> Introduction of Internet of Things (IoT); Need of IoT Security; Requirement and Basic Properties; Main Challenges: confidentiality, integrity, availability and non-repudiation; IoT Architectures (Device, Cloud, Gateway, Backend, Applications); Security Classification &amp; Access Control; Data classification: Public and Private; Privacy issues in IoT; IoT Authentication and Authorization; IoT Data Integrity; Web Based Attacks and Implementation in IoT; Denial of Service; Sniffing, Phishing, DNS Hijacking, Pharming and Defacement; Cryptology Cipher; Symmetric Key Algorithms (AES and DES); Asymmetric Key Algorithm (RSA); Attacks: Dictionary and Brute Force, Lookup Tables, Reverse Lookup Tables and Rainbow Tables; Attack Surface in IoT and Threat Assessment; Embedded Device: UART, SPI, I2C, JTAG; Software and Cloud components: Firmware of the device; Web Application Dashboard; Mobile Application used to Control; Configure and Monitor the devices; Radio Communication: WIFI, BLE, Cellular, ZigBee*, ZWave* and 6LoWPAN; IoT Protocol inbuilt Security Features on Transport Layer and Application Layer; Security Management; Identity and Access Management; Key Management and Model Discussions: Smart Home, Smart Agriculture, Smart Retail Supply, Smart Healthcare, Smart Grid, and Smart Cities;</p> <p><b>Practical</b> Developing Internet of Things &amp; Logical Design using Python: Introduction, IoT Design Methodology, Installing Python, Python Data Types &amp; Data Structures, Control Flow, Functions, Modules, Packages, File Handling, Date/Time Operations, Classes and Python Packages; IoT Physical Devices &amp; Endpoints: What is an IoT Device, Exemplary Device, Board, Linux on Raspberry Pi, Interfaces, and Programming and IoT Devices.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Hanes, D., G. Salgueiro, R. Barton, J. Henry and P. Grossetete. 2017. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things. Cisco Press. Indianapolis, Indiana, USA.</li> </ol>
---	--

<ol style="list-style-type: none"> <li>4. Lea, P. 2018. Internet of Things for Architects: Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security. Packt Publishing, Birmingham, UK.</li> <li>5. Sinclair, B. 2017. How Your Company Can Use the Internet of Things to Win in the Outcome Economy. McGraw-Hill, New York, USA.</li> </ol>	<ol style="list-style-type: none"> <li>2. Khan, J.Y. and M.R. Yuce. 2019. Internet of Things (iot). Pan Stanford Publishing Pte Limited, Central Tower, Temasek Avenue, Singapore.</li> <li>3. Kranz, M. 2016. Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry. John Wiley &amp; Sons, Inc., Hoboken, New Jersey, USA.</li> <li>4. Lea, P. 2018. Internet of Things for Architects: Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security. Packt Publishing, Birmingham, UK.</li> <li>5. Sinclair, B. 2017. How Your Company Can Use the Internet of Things to Win in the Outcome Economy. McGraw-Hill, New York, USA.</li> </ol>
<p><b>IT-512</b>                      <b>Mobile Application Development</b>                      <b>3(2-1)</b></p>	<p><b>IT-512</b>                      <b>MOBILE APPLICATION DEVELOPMENT</b>                      <b>3(2-1)</b></p>
<p><b>Objectives:</b> This course aims at educating students on different architectures &amp; framework for Mobile Application development, learning how to develop mobile applications using current software development environments and comparing different Architectures and Mobile Operating Systems</p> <p><b>Theory:</b> Theory: Mobile Operating Systems; Mobile Development Concepts; Types of Mobile Applications; Technologies Involved in Application Development; Mobile Platform Architecture; Mobile Application Development Tools; Comparison and limitations of popular Cross platform development tools. Overview of Native Android development, iOS development, Web based Hybrid application development using Apache Cordova and Ionic, Native Cross platform Application development using React Native and Flutter, User Interface Concepts, Guidelines and Practices for Improving User Experience; Data Storage and Content Providers; Creating Consumable Web Services for Mobile Devices, Application Resource Management, Network Integration in Mobile Environments, Accessing Mobile device Hardware resources and sensors, Mobile phone camera and their innovative uses, Mobile devices as IOT hub, Cloud and Mobile devices; AI and machine learning support in Mobile Operating Systems; New trends in Mobile Software and Hardware.</p> <p><b>Practical:</b> Installing and Configuring the Mobile Development Environments; Software Development Kit Managers Using mobile platform APIs, UWP, Apple iOS &amp; Android; Comparison and limitations of popular Cross platform development tools; Calling Built-In Applications Using Intents, Displaying and Listening to Notifications. Components of a Screen, User Interface (UI) and User Experience (UX), Adapting and Managing Changes to Screen</p>	<p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Demonstrate different architectures &amp; framework for Mobile Application development.</li> <li>2. Apply the knowledge to develop mobile applications using current software development environments and comparing different Architectures and Mobile Operating Systems</li> <li>3. Explain the understand android SDK</li> <li>4. Code to design apps using Android Studio development tool.</li> </ol> <p><b>Theory</b> Theory: Mobile Operating Systems; Mobile Development Concepts; Types of Mobile Applications; Technologies Involved in Application Development; Mobile Platform Architecture; Mobile Application Development Tools; Comparison and limitations of popular Cross platform development tools. Overview of Native Android development, iOS development, Web based Hybrid application development using Apache Cordova and Ionic, Native Cross platform Application development using React Native and Flutter, User Interface Concepts, Guidelines and Practices for Improving User Experience; Data Storage and Content Providers; Creating Consumable Web Services for Mobile Devices, Application Resource Management, Network Integration in Mobile Environments, Accessing Mobile device Hardware resources and sensors, Mobile phone camera and their innovative uses, Mobile devices as IOT hub, Cloud and Mobile devices; AI and machine learning support in Mobile Operating Systems; New trends in Mobile Software and Hardware.</p> <p><b>Practical</b></p>

<p>Orientation, Utilizing the Action Bar; Utilizing web and cloud services; Rest API's, Views, User Preferences. Sharing Data; Accessing and using various on-board sensors and hardware; Maps and GIS features; Native Machine Learning Hardware Support and Libraries; Mobile Application Deployment.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Burton, M., &amp; Felker, D., 2015. Android App Development For Dummies. John Wiley &amp; Sons.</li> <li>2. Eisenman, B., 2015. Learning react native: building native mobile apps with JavaScript. O'Reilly Media Inc.</li> <li>3. Griffiths, D., &amp; Griffiths, D., 2017. Head First Android Development: a brain-friendly guide. O'Reilly Media Inc.</li> <li>4. Iversen, J., &amp; Eierman, M., 2017. Mobile App Development for iOS and Android, Edition 2.0, Prospect Press.</li> </ol>	<p>Installing and Configuring the Mobile Development Environments; Software Development Kit Managers Using mobile platform APIs, UWP, Apple iOS &amp; Android; Comparison and limitations of popular Cross platform development tools; Calling Built-In Applications Using Intents, Displaying and Listening to Notifications. Components of a Screen, User Interface (UI) and User Experience (UX), Adapting and Managing Changes to Screen Orientation, Utilizing the Action Bar; Utilizing web and cloud services; Rest API's, Views, User Preferences. Sharing Data; Accessing and using various on-board sensors and hardware; Maps and GIS features; Native Machine Learning Hardware Support and Libraries; Mobile Application Deployment.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Burton, M., and Felker, D., 2015. Android App Development For Dummies. John Wiley &amp; Sons.</li> <li>2. Eisenman, B., 2015. Learning react native: building native mobile apps with JavaScript. O'Reilly Media Inc.</li> <li>3. Griffiths, D., and Griffiths, D., 2019. Head First Android Development: a brain-friendly guide. O'Reilly Media Inc.</li> <li>4. Iversen, J., &amp; Eierman, M., 2017. Mobile App Development for iOS and Android, Edition 2.0, Prospect Press.</li> </ol>
<p><b>IT-601</b>                      <b>Human Computer Interaction</b>                      <b>3(2-1)</b></p>	<p><b>IT-601</b>                      <b>HUMAN COMPUTER INTERACTION</b>                      <b>3(2-1)</b></p>
<p><b>Theory:</b> Contexts for HCI; Psychology of usable things; Processes for User-Centered Design; Metrics and Measures for Evaluation; Usability heuristics and principles of Usability testing; Physical capabilities, Cognitive and social models for interaction design; Principles of good interaction design; Accessibility, Principles of GUI; Visual design elements; Data gathering; Task analysis; Prototyping, Help and user documentation; Internationalization; Usability inspection methods; Usability testing methods; New Interaction Technologies; Usability in practice; Visual Design and Typography; Icon Design; Ubiquitous Augmented and Virtual Reality.</p> <p><b>Practical:</b> Task-centered system design: task-centered process, development of task examples, evaluation of designs through a task-centered walk-through; User-centered design and prototyping: assumptions, participatory design, methods for involving the user, prototyping, medium fidelity; Methods for evaluation of interfaces with users: goals, approaches, ethics, introspection, extracting the conceptual model, direct observation, constructive interaction, interviews and questionnaires, continuous evaluation via user feedback and field studies.</p> <p><b>Suggested Readings:</b></p>	<p><b>Course Learning Outcomes</b> By the end of this course, students will be able to:</p> <ol style="list-style-type: none"> <li>1. Explain the concept of human resource management and to understand its relevance in organizations.</li> <li>2. Apply the necessary skill set for application of various HR issues.</li> <li>3. Analyze the strategic issues and strategies required to select and develop manpower resources.</li> <li>4. Implement relevant skills necessary for application in HR related issues</li> <li>5. Integrate the understanding of various HR concepts along with the domain concept in order to take correct business decisions</li> </ol> <p><b>Theory</b> Contexts for HCI; Psychology of usable things; Processes for User-Centered Design; Metrics and Measures for Evaluation; Usability heuristics and principles of Usability testing; Physical capabilities, Cognitive and social models for interaction design; Principles of good interaction design; Accessibility, Principles of GUI; Visual design elements; Data gathering; Task analysis; Prototyping, Help and user documentation; Internationalization; Usability inspection methods; Usability testing methods; New Interaction Technologies; Usability in</p>

<ol style="list-style-type: none"> <li>1. Benyon, D., 2013. Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design. 3<sup>rd</sup> Ed. Pearson, USA.</li> <li>2. Cooper A. 2014. The Essentials of Interaction Design. 4<sup>th</sup> Ed. John Wiley &amp; Sons, USA.</li> <li>3. Sharp, H. and J. Preece. 2019. Interaction Design: Beyond Human-Computer Interaction. 5th Ed. Wiley, USA.</li> <li>4. Shneiderman, B. and C. Plaisant. 2019. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 6th Ed. Pearson, USA.</li> </ol>	<p>practice; Visual Design and Typography; Icon Design; Ubiquitous Augmented and Virtual Reality.</p> <p><b>Practical</b></p> <p>Task-centered system design: task-centered process, development of task examples, evaluation of designs through a task-centered walk-through; User-centered design and prototyping: assumptions, participatory design, methods for involving the user, prototyping, medium fidelity; Methods for evaluation of interfaces with users: goals, approaches, ethics, introspection, extracting the conceptual model, direct observation, constructive interaction, interviews and questionnaires, continuous evaluation via user feedback and field studies.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Benyon, D., 2013. Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design. 3<sup>rd</sup> Ed. Pearson, USA.</li> <li>2. Cooper A. 2014. The Essentials of Interaction Design. 4<sup>th</sup> Ed. John Wiley &amp; Sons, USA.</li> <li>3. Sharp, H. and J. Preece. 2019. Interaction Design: Beyond Human-Computer Interaction. 5th Ed. Wiley, USA.</li> <li>4. Shneiderman, B. and C. Plaisant. 2019. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 6th Ed. Pearson, USA.</li> </ol>
<p><b>IT-603</b>                      <b>System and Network Administration</b>                      <b>3(2-1)</b></p>	<p><b>IT-603</b>                      <b>SYSTEM AND NETWORK ADMINISTRATION</b>                      <b>3(2-1)</b></p>
<p><b>Theory:</b></p> <p>Introduction to System Administration; SA Components; Server Environment (Microsoft and Linux); Reliable Products; Server Hardware Costing; Maintenance Contracts and Spare parts; Maintaining Data Integrity; Client Server OS Configuration; Providing Remote Console Access; Comparative Analysis of OS: Important Attributes, Key Features, Pros and Cons; Linux Installation and Verification; Configuring local services and managing basic system issues; Administer users and groups; Software management. managing network services and network monitoring tools; Boot management and process management; IP tables and filtering; Securing network traffic; Advanced file systems and logs; Bash shell scripting; Configuring servers: FTP, NFS, Samba, DHCP, DNS and Apache.</p> <p><b>Practical:</b></p> <p>Installation of Linux on the machine; Configuration of Microsoft windows and Linux on the server and client machines.</p> <p><b>Suggested Readings:</b></p>	<p><b>Course Learning Outcomes</b></p> <p>By the end of this course, the students will be able to:</p> <ol style="list-style-type: none"> <li>1. Demonstrate the key terminologies and technologies of System and network administration.</li> <li>2. Apply the services and functions provided by different data centers.</li> <li>3. Use various OS used to handle system and network administration routines.</li> </ol> <p><b>Theory</b></p> <p>Introduction to System Administration; SA Components; Server Environment (Microsoft and Linux); Reliable Products; Server Hardware Costing; Maintenance Contracts and Spare parts; Maintaining Data Integrity; Client Server OS Configuration; Providing Remote Console Access; Comparative Analysis of OS: Important Attributes, Key Features, Pros and Cons; Linux Installation and Verification; Configuring local services and managing basic system issues; Administer users and groups; Software management. managing network services and network monitoring tools; Boot management and process management; IP</p>

<ol style="list-style-type: none"> <li>1. Limoncelli, T., C. Hogan and S. Chalup. 2017. The Practice of System and Network Administration. 3rd Ed. Addison-Wesley, USA.</li> <li>2. Chao, L. 2009. Networking Systems Design and Development. CRC Press, Boca Raton, FL, USA.</li> <li>3. VonHagen, W. 2011. Red Hat Enterprise Linux 6 Bible: Administering Enterprise Linux Systems. John Wiley &amp; Sons. NJ, USA.</li> </ol>	<p>tables and filtering; Securing network traffic; Advanced file systems and logs; Bash shell scripting; Configuring servers: FTP, NFS, Samba, DHCP, DNS and Apache.</p> <p><b>Practical</b></p> <p>Installation of Linux on the machine; Configuration of Microsoft windows and Linux on the server and client machines.</p> <p><b>Suggested Readings</b></p> <ol style="list-style-type: none"> <li>1. Limoncelli, T., C. Hogan and S. Chalup. 2018. The Practice of System and Network Administration. 3rd Ed. Addison-Wesley, USA.</li> <li>2. Chao, L. 2009. Networking Systems Design and Development. CRC Press, Boca Raton, FL, USA.</li> <li>3. VonHagen, W. 2011. Red Hat Enterprise Linux 6 Bible: Administering Enterprise Linux Systems. John Wiley &amp; Sons. NJ, USA.</li> </ol>
<p><b>IT-605</b>                      <b>Information Technology Project Management</b>                      <b>3(2-1)</b></p>	<p><b>IT-605</b>                      <b>INFORMATION TECHNOLOGY PROJECT MANAGEMENT</b>                      <b>3(2-1)</b></p>
<p><b>Objectives:</b></p> <p>The objective of this course is to provide a foundation to students as a future IT project manager to understand the job roles of an IT project manager also recognize the key issues during the IT project management procedures and describe the best practices in IT project management processes.</p> <p><b>Theory:</b></p> <p>Project management principles, Project roles and responsibilities, Project definition, The Project Management and Information Technology Context. The Project Management Process Groups. Project Integration Management. Project Scope Management. Project Time Management. Project Cost Management. Project Human Resource Management. Project Communications Management. Project Procurement Management. Project Management Tools, Project planning, estimating and resourcing, Project issue management, Project risk management, Project quality management, Project change management, Project controlling and reporting, IT Infrastructure Library (ITIL), Service Science, Management, and Engineering (SSME), Communication management and team building.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Kathy, S., 2015. Information Technology Project Management, 8th Edition. ISBN-13: 978-1285452340</li> <li>2. Joseph, P., 2010. IT Project Management: On Track from Start to Finish by Joseph Phillips, McGraw Hill Osborne Media; 3rd Edition.</li> <li>3. Jack, T., 2009. Information Technology Project Management; Wiley; 3rd Edition.</li> </ol>	<p><b>Course Learning Outcomes</b></p> <p>By the end of this course, the students will be able to:</p> <ol style="list-style-type: none"> <li>1. Identify IT project manager role to understand the responsibilities also recognize the key issues during the IT project management procedures and describe the best practices in IT project management processes.</li> <li>2. Examine critical thinking skills in the development of complex activities and in the provision of constructive criticism</li> <li>3. Demonstrate fundamental IT methods and simple algorithms in the construction of a real or simulated system solution.</li> <li>4. Demonstrate technical competency in at least two areas of programming, system design, networking, and project management.</li> <li>5. Apply the ability to communicate, in written, visual and verbal form, in order to convey complex information to others in a way that supports decision-making.</li> </ol> <p><b>Theory</b></p> <p>Project management principles, Project roles and responsibilities, Project definition, The Project Management and Information Technology Context. The Project Management Process Groups. Project Integration Management. Project Scope Management. Project Time Management. Project Cost Management. Project Human Resource Management. Project Communications Management. Project Procurement Management. Project Management Tools, Project planning, estimating and resourcing, Project issue management, Project risk management, Project quality management, Project change management, Project controlling</p>

			and reporting, IT Infrastructure Library (ITIL), Service Science, Management, and Engineering (SSME), Communication management and team building.
			<b>Suggested Readings</b>
			1. Kathy, S., 2019. Information Technology Project Management, 8th Edition. ISBN-13: 978-1285452340
			2. Joseph, P., 2010. IT Project Management: On Track from Start to Finish by Joseph Phillips, McGraw Hill Osborne Media; 3rd Edition.
			3. Jack, T., 2009. Information Technology Project Management; Wiley; 3rd Edition.
<b>IT-607</b>	<b>Next Generation Networks</b>	<b>3(2-1)</b>	<b>IT-607</b> <b>NEXT GENERATION NETWORKS</b> <b>3(2-1)</b>
<b>Theory:</b>			<b>Course Learning Outcomes</b>
NGN standardization by ITU; fixed and mobile broadband as basis for NGN, NGN architectures; NGN services; QoS and IMS for NGN deployments; VoIP and IPTV services over NGN; Internet of Everything; VPN in NGN; IPv6-based NGN; migration scenarios from legacy networks to NGN; business approaches and regulation for NGN; future networks as defined by ITU; including network virtualization; software defined networking; smart ubiquitous networks; big data issues; OTT service providers versus Telco service models; impact of M2M in the future; convergence of regulation towards future networks; cloud computing; including ITU's framework; cloud ecosystem; architectures and cloud service models; cloud security; OTT cloud services; Telco cloud implementations; mobile cloud computing services and applications; as well as business and regulation aspects for cloud computing.			By the end of this course, the students will be able to:
<b>Practical:</b>			1. Explain high-quality broadband communications by controlling the quality of service (QoS) on an end-to-end basis.
Implementation of NGN protocols regarding Packet Switching (ATM, IP, MPLS, Ethernet IP, IPv6, VLANs); Virtual Private Networks; Wireless protocols; Inter-VLAN Routing and Gateway Redundancy; Multi-Media System (VoIP, SIP, Video over IP); IMS systems & protocols, Security using network networking simulation tools and packet tracer.			2. Demonstrate multiple QoS classes having different features and to enable the appropriate QoS class to be selected when an application is provided.
<b>Suggested Readings:</b>			3. Apply knowledge to implement software based switches
1. Zhu, H., D. Niyato, W. Saad, and T. Başar, 2019. Game Theory for Next Generation Wireless and Communication Networks: Modeling, Analysis, and Design			<b>Theory</b>
2. Wilkinson, N. 2002. Next generation networks services: Technologies and strategies. John Wiley & Sons, West Sussex, England.			NGN standardization by ITU; fixed and mobile broadband as basis for NGN, NGN architectures; NGN services; QoS and IMS for NGN deployments; VoIP and IPTV services over NGN; Internet of Everything; VPN in NGN; IPv6-based NGN; migration scenarios from legacy networks to NGN; business approaches and regulation for NGN; future networks as defined by ITU; including network virtualization; software defined networking; smart ubiquitous networks; big data issues; OTT service providers versus Telco service models; impact of M2M in the future; convergence of regulation towards future networks; cloud computing; including ITU's framework; cloud ecosystem; architectures and cloud service models; cloud security; OTT cloud services; Telco cloud implementations; mobile cloud computing services and applications; as well as business and regulation aspects for cloud computing.
3. Ohrtman, F.D. 2003. Softswitch Architecture for VoIP. 1st Ed. McGraw-Hill Professional, USA.			<b>Practical</b>
4. Mueller, S M. 2002. APIs and Protocols for Convergent Network Services. McGraw-Hill Professional, NY, USA.			Implementation of NGN protocols regarding Packet Switching (ATM, IP, MPLS, Ethernet IP, IPv6, VLANs); Virtual Private Networks; Wireless protocols; Inter-VLAN Routing and Gateway Redundancy; Multi-Media System (VoIP, SIP, Video over IP); IMS systems & protocols, Security using network networking simulation tools and packet tracer.
			<b>Suggested Readings</b>
			1. Hanrahan, H. 2007. Network Convergence: services, Applications, Transport, and Operations Support. John Wiley & Sons, West Sussex, England.



5. Hanrahan, H. 2007. Network Convergence: services, Applications, Transport, and Operations Support. John Wiley & Sons, West Sussex, England.	2. Mueller, S M. 2002. APIs and Protocols for Convergent Network Services. McGraw-Hill Professional, NY, USA. 3. Ohrtman, F.D. 2003. Softswitch Architecture for VoIP. 1st Ed. McGraw-Hill Professional, USA. 4. Wilkinson, N. 2002. Next generation networks services: Technologies and strategies. John Wiley & Sons, West Sussex, England. 5. Zhu, H., D. Niyato, W. Saad, and T. Başar, 2019. Game Theory for Next Generation Wireless and Communication Networks: Modeling, Analysis, and Design
<b>IT-609</b> <b>Research Report</b> <b>2(0-2)</b>	<b>IT-609</b> <b>FINAL YEAR PROJECT-I</b> <b>2(0-2)</b>
<b>Practical:</b> Abstract; Introduction; Materials and methods; Literature review; Results and Discussion; Summary of the report by the help of Requirement gathering techniques, analysis, designing of real time application/system related to any enterprise; record of a sequence of events, interpretation of the significance of these events or facts; evaluation of the facts or results of research; discussion of the outcomes of a decision or course of action. <b>Suggested Readings:</b> <ol style="list-style-type: none"> <li>1. Arosteguy, K. O., Bright, A. and Rinard, B. J. 2019. A Student's Guide to Academic and Professional Writing in Education. Teachers College Press, USA.</li> <li>2. Campbell, A. 2019. Agile Project Management with Scrum: Secret Scrum Formulas and Methods in Agile Project Management. Addison-Wesley Professional, USA.</li> <li>3. Carlock, J., Eberhardt, M., Horst, J., &amp; Kolenich, P. 2017. The International Student's Guide to Writing a Research Paper. University of Michigan Press, USA.</li> <li>4. Wang, G. T. and Park, K. 2016. Student research and report writing: From topic selection to the complete paper. John Wiley &amp; Sons, USA.</li> </ol>	<b>Course Learning Outcomes</b> By the end of this course, the students will be able to: <ol style="list-style-type: none"> <li>1. Demonstrate conceptual perception of project that they will be submitted.</li> <li>2. Demonstrate backend support, algorithms, and required tools.</li> <li>3. Explain complete knowledge of materials and methods.</li> </ol>
<b>IT-608</b> <b>Internship</b> <b>2(0-2)</b>	<b>IT-608</b> <b>INTERNSHIP</b> <b>2(0-2)</b>
<b>IT-610</b> <b>Final Year Project</b> <b>4(0-4)</b>	<b>IT-610</b> <b>FINAL YEAR PROJECT-II</b> <b>4(0-4)</b>

<p><b>Practical:</b>  Problem Statement; Overall Goals of Product/Service/Program; Outcomes (or client/customer impacts) and Performance; Overall Evaluation Goals (what questions are being answered by the research?) Methodology; Types of data/information; How data/information were collected (what instruments were used, etc.); How data/information were analyzed, Limitations of the evaluation (cautions about findings/conclusions and how to use the findings/conclusions. Requirement gathering techniques, analysis; Designing; development and implementation of real time application/system related to any enterprise.</p> <p><b>Suggested Readings:</b></p> <ol style="list-style-type: none"> <li>1. Campbell, A. 2019. Agile Project Management with Scrum: Secret Scrum Formulas and Methods in Agile Project Management. Addison-Wesley Professional, USA.</li> <li>2. Cervantes, H. and Kazman, R. 2016. Designing software architectures: a practical approach. Addison-Wesley Professional, USA.</li> <li>3. Robert C. 2017. Clean Architecture: A Craftsman's Guide to Software Structure and Design, Prentice Hall, USA.</li> <li>4. Wang, G. T. and Park, K. 2016. Student research and report writing: From topic selection to the complete paper. John Wiley &amp; Sons, USA.</li> </ol>	<p><b>Course Learning Outcomes</b>  By the end of this course, the students will be able to:</p> <ol style="list-style-type: none"> <li>1. Justify layout of the project conceived in previous semester.</li> <li>2. Illustrate process model, architecture, and layout of the completed project.</li> <li>3. Explain program code and flow diagrams.</li> <li>4. Write scientifically complete project documentation explaining all aspects of the completed project.</li> </ol>
---	---