Dr. Babasaheb Ambedkar Technological University

(Established as a University of Technology in the State of Maharashtra)
(Under Maharashtra Act No. XXIX of 2014)
P.O. Lonere, Dist. Raigad, Pin- 402 103, Maharashtra
Telephone and Fax. : 02140 - 275142
www.dbatu.ac.in

Course Structure and Detailed Syllabus
for

Second Year, Third Year and Final Year
B. Tech. Programme in Information Technology
(Effective from Academic Year 2020-21)

Programme Objectives:

The program educational objectives for the B. Tech. programme in Information Technology describes accomplishments that graduates are expected to attain within the four years of graduation. Graduates will be able to apply their expertise to contemporary problem solving, be engaged professionally, and have continued to learn and adapt, and have contributed to their organizations through leadership and teamwork. More specifically, the objectives are:

1. PEO1: To enable graduates gain strong skills for employment in multidisciplinary domains driven by IT
2. PEO2: To enable graduates to pursue higher education and research
3. PEO3: To enable graduates to develop entrepreneurship and leadership skills
4. PEO4: To enable graduates to contribute to the society in accordance with highest standards of ethics
5. PEO5: To develop breakthrough solutions enabling transformations in a rapidly changing IT world

Programme Outcomes:

The graduates of this programme will be able to demonstrate:

1. PO1: An Understanding of IT architecture, software and hardware concepts, functionalities and applications
2. PO2: An Ability to design, develop and test computer programs involving various algorithms, methodology and programming languages
3. PO3: Competency of business domains and functional processes that employ IT systems and applications
4. PO4: Practical use of communication protocols and their applications in the field of Internet and World Wide Web
5. PO5: Sound understanding of fundamentals of computer as the central enabling platform for information management in 21st century
6. PO6: An Ability to develop, integrate, maintain and innovate software applications deployed in various multi-disciplinary domains
7. PO7: Thought leadership to design and implement practical solutions for global industry needs.
8. PO8: Acumen to embrace and adopt futuristic IT technological developments
9. PO9: Sound knowledge of entrepreneurship traits to succeed
10. PO10: Adoption of practices that are ethical ensuring transparency and accountability
11. PO11: Capability to provide solutions that are socially empowering and environment friendly
12. PO12: Effective communication and collaboration techniques with stakeholders to achieve best results.

Dr. Babasaheb Ambedkar Technological University

Department of Information Technology

Basic Science Core (BSC)			
BTBSC301	Engineering Mathematics - III	(3-1-0)	4

Engineering Science Core (ESC)			
BTITC303	Discrete Mathematics	(3-1-0)	4

Humanities and Social Science Core (HSC)			
BTHM3402	Interpersonal Communication Skills and Self Development for Engineers	(2-0-0)	2
BTHM501	Constitution of India	(2-0-0)	Audit
BTCOE406C	Soft Skills and personality developement	(2-0-0)	Audit
BTHM701	Essence of Indian Tradition	(2-0-0)	Audit

Program Core (PC)			
BTITC304	Object Oriented Paradigm with C++	(3-1-0)	4
BTITC305	Data Structures and Applications	(3-1-0)	4
BTITL306	Object Oriented Paradigm with C++ and Data Structures and Applications Lab	(0-0-4)	2
BTITC401	Probability and Statistics	(3-1-0)	4
BTITC402	Computer Architecture and Organization	(3-1-0)	4
BTITC403	Design and Analysis of Algorithms	(3-1-0)	4
BTITC404	Organizational Behavior	(3-0-0)	3
BTITL406	Design and Analysis of Algorithms and Elective- I Lab	(0-0-4)	2
BTITC501	Software Engineering	(3-1-0)	4
BTITC502	Computer Networks and Internetworking Protocols	(3-1-0)	4
BTITL505	Computer Networks and Internetworking Protocols Lab	(0-0-2)	1
BTITC601	Operating Systems	(3-1-0)	4
BTITC602	Database Management Systems	(3-1-0)	4
BTITL605	DBMS Lab	(0-0-2)	1
BTITL606	Operating Systems Lab	(0-0-2)	1
BTITC701	Machine Learning	(3-1-0)	4
BTITL705	ML Lab	(0-0-2)	1
BTITL706	Elective- VI and Elective- VII Lab	(0-0-4)	2

Program Specific Electives (PSE)			
Elective-I			
BTITPE405A	Microprocessors and Microcontrollers	(2-1-0)	3
BTITPE405B	Web Technology		
BTITPE405C	Physics of Engineering Materials		
Elective- II			
BTITPE503A	Embedded Systems	(3-0-0)	3
BTITPE503B	IT Service Management		
BTITPE503C	Information Storage Management		
BTITPE503D	Network Management		
BTITPE503E	Data Visualization		
Elective- III			
BTITOE504A	Theory of Computation	(3-0-0)	3
BTITOE504B	Graph Theory		
BTITOE504C	Programming in Java		
BTITOE504D	Human Computer Interaction		
BTITOE504E	Game Theory		
BTITL506	SE and Elective- II Lab	(0-0-4)	2
Elective- IV			
BTITPE603A	Software Testing	(3-0-0)	3
BTITPE603B	Data Storage Technologies & Networks		
BTITPE603C	Service Oriented Architecture		
BTITPE603D	Network Programming		
BTITPE603E	Data Warehousing & Data Mining		
Elective- V			
BTITOE604A	Compiler Design	(3-0-0)	3
BTITOE604B	Enterprise Resource Planning		
BTITOE604C	Decision Support Systems		
BTITOE604D	Software Project Management		
BTITL606	Elective-IV Lab	(0-0-2)	1
Elective- VI			
BTITPE702A	Pattern Recognition	(3-0-0)	3
BTITPE702B	Soft Computing		
BTITPE702C	Artificial Neural Networks		
Elective- VII			
BTITPE703A	Real Time Systems	(3-0-0)	3
BTITPE703B	Information Security		
BTITPE703C	Electronic Payment Systems		
BTITPE703D	Distributed Computing		
BTITPE703E	GPU Computing		

Elective- VIII			
BTITOE704A	Natural Language Processing	(3-0-0)	3
BTITOE704B	Artificial Intelligence		
BTITOE704C	Advanced Database Technology		

Mandatory Learning Courses (MLC)			
BTITS307	Seminar - I	(0-0-0)	2
Internship - I Evaluation	Field Training / Internship/Industrial Training (minimum of 4 weeks which can be completed partially in third semester and fourth semester or at one time).	(0-0-0)	Audit
BTITS407	Seminar - II	(0-0-0)	2
Internship - II	Field Training / Internship/Industrial Training (minimum of 4 weeks which can be completed partially in third semester and fourth semester or at one time).	(0-0-0)	To be audited in V Sem.
BTITF508	Internship – II Evaluation	(0-0-0)	Audit
Internship- III	Field Training / Internship/Industrial Training (minimum of 4 weeks which can be completed partially in fifth semester and sixth semester or at one time).	(0-0-0)	To be audited in VII Sem.
BTITF708	Internship – III Evaluation	(0-0-0)	Audit

Project(MP)			
BTITP507	Mini Project - I	(0-0-4)	4
BTITP607	Mini Project - II	(0-0-4)	4
BTITP707	Mini Project - III	(0-0-4)	4

Major Project			
BTITP801	Project work/ Internship	(0-0-0)	12

Honors courses(for B.tech programme in I.T.)

BTITH01	Internet of Things	(4-0-0)	4
BTITH02	Big Data Computing		
BTITH03	Block Chain Technology		
BTITH04	Data Science		
BTITH05	Computer Vision		
BTITH06	Advanced Machine Learning		
BTITH07	Recommender Systems		
BTITH08	Reinforcement Learning		

Add on courses

BTITAO01	Database Programming	(3-0-0)	3
BTITAO02	Deep Neural Networks	(3-0-0)	3
BTITAO03	Compliance, Risk and Governance: IT Perspective	(3-0-0)	3
BTITAO04	ERP: Oracle Applications	(3-0-0)	3
BTITAOL01	Database Programming Lab	(0-0-2)	1
BTITAOL02	Deep Neural Networks Lab	(0-0-2)	1
BTITAOL03	Compliance, Risk and Governance: IT Perspective Lab	(0-0-2)	1
BTITAOL04	ERP: Oracle Applications Lab	(0-0-2)	1

B. Tech. Programme in Information Technology

Second Year B. Tech. (With effect from 2021-22)

Semester III										
Course Category	**Course Code**	**Course Title**	**Teaching Scheme**			**Evaluation Scheme**				
			L	T	P	CA	MSE	ESE	Total	Credit
BSC	BTBSC301	Engineering Mathematics - III	3	1	-	20	20	60	100	4
HSSMC	BTHM3402	Interpersonal Communication Skills and Self Development for Engineers	2	-	-	20	20	60	100	2
PCC	BTITC303	Discrete Mathematics	3	1	-	20	20	60	100	4
PCC	BTITC304	Object Oriented Paradigm with C++	3	1	-	20	20	60	100	4
PCC	BTITC305	Data Structures and Applications	3	1	-	20	20	60	100	4
LC	BTITL306	Object Oriented Paradigm with C++, Data Structures and Applications Lab	-	-	4	60	-	40	100	2
Seminar	BTITS307	Seminar - I	-	-	-	60	-	40	100	2
Internship	Internship - I	Internship - I Evaluation	-	-	-	-	-	-	-	Audit
			14	**4**	**4**	**220**	**100**	**380**	**700**	**22**
	Semester IV									
PCC	BTITC401	Probability and Statistics	3	1	-	20	20	60	100	4
PCC	BTITC402	Computer Architecture and Organization	3	1	-	20	20	60	100	4
PCC	BTITC403	Design and Analysis of Algorithms	3	1	-	20	20	60	100	4
HSSMC	BTITC404	Organizational Behavior	3	-	-	20	20	60	100	3
PEC		**Elective-I**								
	BTITPE405A	Microprocessors and Microcontrollers	2	1	-	20	20	60	100	3
	BTITPE405B	Web Technology								
	BTITPE405C	Physics of Engineering Materials								
HSSMC	BTCOE406C	Soft Skills and Personality Development	2	-	-	-	-	50	50	Audit
LC	BTITL406	Design and Analysis of Algorithms, Elective- I Lab	-	-	4	60	-	40	100	2
Seminar	BTITS407	Seminar - II	-	-	-	60	-	40	100	2
Internship	Internship - II	Field Training / Internship/Industrial Training (minimum of 4 weeks which can be completed partially in third semester and fourth semester or at one time).	-	-	-	-	-	-	-	To be audited in V Sem.
			14	**4**	**4**	**220**	**100**	**380**	**700**	**22**

BSC = Basic Science Course, ESC = Engineering Science Course, PCC = Professional Core Course
PEC = Professional Elective Course, OEC = Open Elective Course, LC = Laboratory Course
HSSMC = Humanities and Social Science including Management Courses

Third Year B. Tech. (With effect from 2022-23)

	Semester V									
Course Category	**Course Code**	**Course Title**	**Teaching Scheme**			**Evaluation Scheme**				
			L	T	P	CA	MSE	ESE	Total	Credits
PCC	BTITC501	Software Engineering	3	1	-	20	20	60	100	4
PCC	BTITC502	Computer Networks and Internetworking Protocols	3	1	-	20	20	60	100	4
PEC		**Elective- II**	3	-	-	20	20	60	100	3
	BTITPE503A	Embedded Systems								
	BTITPE503B	IT Service Management								
	BTITPE503C	Information Storage Management								
	BTITPE503D	Network Management								
	BTITPE503E	Data Visualization								
OEC		**Elective- III**	3	-	-	20	20	60	100	3
	BTITOE504A	Theory of Computation								
	BTITOE504B	Graph Theory								
	BTITOE504C	Programming in Java								
	BTITOE504D	Human Computer Interaction								
	BTITOE504E	Game Theory								
HSSMC	BTHM501	Constitution of India	2	-	-	-	-	-	-	Audit
LC	BTITL505	Computer Networks and Internetworking Protocols Lab	-	-	2	60	-	40	100	1
LC	BTITL506	Software Engineering and Elective- II Lab	-	-	4	60	-	40	100	2
Project	BTITP507	Mini Project - I	-	-	4	60	-	40	100	4
Internship	BTITF508	Internship – II Evaluation	-	-	-	-	-	-	-	Audit
			14	**2**	**10**	**260**	**80**	**360**	**700**	**21**
Semester VI										
PCC	BTITC601	Operating Systems	3	1	-	20	20	60	100	4
PCC	BTITC602	Database Management Systems	3	1	-	20	20	60	100	4
PEC		**Elective- IV**	3	-	-	20	20	60	100	3
	BTITPE603A	Software Testing								
	BTITPE603B	Data Storage Technologies & Networks								
	BTITPE603C	Service Oriented Architecture								
	BTITPE603D	Network Programming								
	BTITPE603E	Data Warehousing & Data Mining								
OEC		**Elective- V**	3	-	-	20	20	60	100	3
	BTITOE604A	Compiler Design								
	BTITOE604B	Enterprise Resource Planning								
	BTITOE604C	Decision Support Systems								
	BTITOE604D	Software Project Management								

LC	BTITL605	Database Management Systems Lab	-	-	2	60	-	40	100	1
LC	BTITL606	Operating Systems and Elective-IV Lab	-	-	4	60	-	40	100	2
Project	BTITP607	Mini Project - II	-	-	4	60	-	40	100	4
Internship	Internship- III	Field Training / Internship/Industrial Training (minimum of 4 weeks which can be completed partially in fifth semester and sixth semester or at one time).	-	-	-	-	-	-	-	To be audited in VII Sem.
			12	**2**	**10**	**260**	**80**	**360**	**700**	**21**

BSC = Basic Science Course, ESC = Engineering Science Course, PCC = Professional Core Course
PEC = Professional Elective Course, OEC = Open Elective Course, LC = Laboratory Course
HSSMC = Humanities and Social Science including Management Courses

Add on courses offered by the department

Sr. No.	**Add-on Courses****		**Teaching Scheme**			**Evaluation Scheme**			**Total Marks**	**Credits**	**Total Hours**
			L	T	P	MSE	CA	ESE			
								Internal / External			
1	BTITAO01	Database Programming	3	-	-	20	20	60	100	3	3
2	BTITAO02	Deep Neural Networks	3	-	-	20	20	60	100	3	3
3	BTITAO03	Compliance, Risk and Governance: IT Perspective	3	-	-	20	20	60	100	3	3
4	BTITAO04	ERP: Oracle Applications	3	-	-	20	20	60	100	3	3
5	BTITAOL01	Database Programming Lab	-	-	2	-	60	40	100	1	2
6	BTITAOL02	Deep Neural Networks Lab	-	-	2	-	60	40	100	1	2
7	BTITAOL03	Compliance, Risk and Governance: IT Perspective Lab	-	-	2	-	60	40	100	1	2
8	BTITAOL04	ERP: Oracle Applications Lab	-	-	2	-	60	40	100	1	2

**Add-on course will be offered to fifth and seventh semester students for which additional credits will be given.

Final Year B. Tech. (With effect from 2023-24)

	Semester VII									
Course Category	**Course Code**	**Course Title**	**Teaching Scheme**			**Evaluation Scheme**				
			L	T	P	CA	MSE	ESE	Total	Credit
PCC	BTITC701	Machine Learning	3	1	-	20	20	60	100	4
PEC	BTITPE702A BTITPE702B BTITPE702C	**Elective- VI** Pattern Recognition Soft Computing Artificial Neural Networks	3		-	20	20	60	100	3
PEC	BTITPE703A BTITPE703B BTITPE703C BTITPE703D BTITPE703E	**Elective- VII** Real Time Systems Information Security Electronic Payment Systems Distributed Computing GPU Computing	3	-	-	20	20	60	100	3
OEC	BTITOE704A BTITOE704B BTITOE704C	**Elective- VIII** Natural Language Processing Artificial Intelligence Advanced Database Technology	3	-	-	20	20	60	100	3
LC	BTITL705	Machine Learning Lab	-	-	2	60	-	40	100	1
LC	BTITL706	Elective- VI and Elective- VII Lab	-	-	4	60	-	40	100	2
Project	BTITP707	Mini Project - III	-	-	4	60	-	40	100	4
Internship	BTITF708	Internship – III Evaluation	-	-	-	-	-	-	-	Audit
			12	**1**	**10**	**260**	**80**	**360**	**700**	**20**
	Semester VIII									
Project/ Internship	BTITP801	Project work/ Internship	-	-	-	60	--	40	100	12
			-	**-**	**-**	**60**		**40**	**100**	**12**

BSC = Basic Science Course, ESC = Engineering Science Course, PCC = Professional Core Course
PEC = Professional Elective Course, OEC = Open Elective Course, LC = Laboratory Course
HSSMC = Humanities and Social Science including Management Courses

Add on courses offered by the department

Sr. No.	Add-on Courses**		Teaching Scheme			Evaluation Scheme				Total Marks	Credits	Total Hours
			L	T	P	MSE	CA	ESE				
								Internal	External			
1	BTITAO01	Database Programming	3	-	-	20	20	60		100	3	3
2	BTITAO02	Deep Neural Networks	3	-	-	20	20	60		100	3	3
3	BTITAO03	Compliance, Risk and Governance: IT Perspective	3	-	-	20	20	60		100	3	3
4	BTITAO04	ERP: Oracle Applications	3	-	-	20	20	60		100	3	3
5	BTITAOL01	Database Programming Lab	-	-	2	-	60	40		100	1	2
6	BTITAOL02	Deep Neural Networks Lab	-	-	2	-	60	40		100	1	2
7	BTITAOL03	Compliance, Risk and Governance: IT Perspective Lab	-	-	2	-	60	40		100	1	2
8	BTITAOL04	ERP: Oracle Applications Lab	-	-	2	-	60	40		100	1	2

**Add-on course will be offered to fifth and seventh semester students for which additional credits will be given.

Honors courses

Sr. No.	Honors Courses (For B.Tech. Programme in IT; any five Courses)		Teaching Scheme			Evaluation Scheme				Total Marks	Credits	Total Hours
			L	T	P	MSE	CA	ESE				
								Internal	External			
1	BTITH01	Internet of Things	3	-	-	20	20	60		100	3	3
2	BTITH02	Big Data Computing	3	-	-	20	20	60		100	3	3
3	BTITH03	Block Chain Technology	3	-	-	20	20	60		100	3	3
4	BTITH04	Data Science	3	-	-	20	20	60		100	3	3
5	BTITH05	Computer Vision	3	-	-	20	20	60		100	3	3
6	BTITH06	Advanced Machine Learning	3	-	-	20	20	60		100	3	3
7	BTITH07	Recommender Systems	3	-	-	20	20	60		100	3	3
8	BTITH08	Reinforcement Learning	3	-	-	20	20	60		100	3	3

Minor Courses (For other B.Tech. Programme; any Five Courses)

1. Computer Architecture and Organization
2. Data Structures and Applications
3. Database Management Systems
4. Operating Systems
5. Software Engineering
6. Machine Learning
7. Computer Networks and Internetworking Protocols
8. Web Technology

Course Title:	**Engineering Mathematics – III**	**Semester III**	
Course Code	**BTBS301**	**Course Type**	**Compulsory**
Prerequisite	**Engineering Mathematics – II**	**L – T – P**	**3 – 1 – 0**
Stream	**Basic Science**	**Credits**	**4**

Course Objectives:

1. To provide in depth knowledge of complex numbers
2. To find the solution of differential equations
3. To find an in-depth knowledge of Fourier series analysis of periodic function

Course Outcomes:

After learning the course the students should be able:

1. To develop an ability to use characteristics of complex numbers in problem pertaining to electric circuits
2. To develop an acquaintance with the method of finding solution of differential equations
3. To develop an in depth knowledge of vector differentiation and vector integration
4. To develop Fourier series expansion of different periodic functions

Course Content:

UNIT I

Laplace Transform: Definition – conditions for existence ; Transforms of elementary functions ; Properties of Laplace transforms - Linearity property, first shifting property, second shifting property, transforms of functions multiplied by t^n, scale change property, transforms of functions divided by t, transforms of integral of functions, transforms of derivatives ; Evaluation of integrals by using Laplace transform ; Transforms of some special functions- periodic function, Heaviside-unit step function, Dirac delta function.

UNIT II

Inverse Laplace Transform: Introductory remarks ; Inverse transforms of some elementary functions ; General methods of finding inverse transforms ; Partial fraction method and Convolution Theorem for finding inverse Laplace transforms ; Applications to find the solutions of linear differential equations and simultaneous linear differential equations with constant coefficients.

UNIT III

Fourier Transform: Definitions – integral transforms ; Fourier integral theorem (without proof) ; Fourier sine and cosine integrals ;Complex form of Fourier integrals ; Fourier sine and cosine transforms ; Properties of Fourier transforms; Parseval's identity for Fourier Transforms.

UNIT IV

Partial Differential Equations and Their Applications: Formation of Partial differential equations by eliminating arbitrary constants and functions; Equations solvable by direct integration; Linear equations of first order (Lagrange's linear equations); Method of separation of variables – applications to find solutions of one dimensional heat flow equation $\left(\frac{\partial u}{\partial t}=c^2\frac{\partial^2 u}{\partial x^2}\right)$, and one dimensional wave equation (i.e. $\frac{\partial^2 y}{\partial t^2}= c^2\frac{\partial^2 y}{\partial x^2}$).

UNIT V

Functions of Complex Variables: Analytic functions; Cauchy- Riemann equations in Cartesian and polar forms; Harmonic functions in Cartesian form, Cauchy's integral theorem; Cauchy's integral formula; Residues; Cauchy's residue theorem (All theorems without proofs).

Text Books:

1. B. S. Grewal, **"*Higher Engineering Mathematics*"**, Khanna Publishers, New Delhi.
2. H. K. Das, Er. RajnishVerma, **"*Higher Engineering Mathematics*"**, S. Chand & CO. Pvt. Ltd., New Delhi.
3. Dr. B. B. Singh, **"A course in Engineering Mathematics (Volume-III)"**, Synergy Knowledge ware, Mumbai.
4. B. V. Ramana, **"*Higher Engineering Mathematics"*,** Tata McGraw-Hill Publications, New Delhi.

Reference Books:

1. Erwin Kreyszig, **"*Advanced Engineering Mathematics*"**, John Wiley & Sons, New York.
2. Peter O' Neil, ***"A Text Book of Engineering Mathematics"***, Thomson Asia Pvt. Ltd., Singapore.
3. C. R. Wylie, L. C. Barrett, ***"Advanced Engineering Mathematics"***, Tata McGraw-Hill Publishing Company Ltd., New Delhi.
4. C. R. Wylie & L. C. Barrett, ***"Integral Transforms and their Engineering Applications"***, Synergy Knowledge ware, Mumbai.
5. I. N. Sneddon, ***"Integral Transforms"***, Tata McGraw-Hill, New York.

General Instructions:

1. The tutorial classes in Engineering Mathematics-III are to be conducted batch wise. Each class should be divided into three batches for the purpose.
2. The internal assessment of the students for 20 marks will be done based on assignments, surprise tests, quizzes, innovative approach to problem solving and percentage attendance.
3. The minimum number of assignments should be eight covering all topics.

Course Title:	**Interpersonal Communication Skills and Self Development for Engineers**	**Semester III**	
Course Code	**BTHM3402**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**2 – 0 - 0**
Stream	**Humanities, Social Science and Management**	**Credits**	**2**

Course Objectives:

1. To build the skills like team building so that they can work efficiently in groups.
2. To provide knowledge of conflict management while working in large organizations.
3. To develop management skills required in routine work environment.
4. To polish the personality of the learners in order to make them good leaders and employees.
5. To imbibe qualities like manners & etiquettes coordination, mutual understanding for their colleague while working in group.

Course Outcomes:

1. Learners will acquire interpersonal communication skills.
2. Learners will develop the ability to work independently.
3. Learners will develop the qualities like self-discipline, self-criticism and self-management.
4. Learners will have the qualities of time management and discipline.
5. Learners would be able to present themselves as an inspiration for others.

UNIT I

Development of Proficiency in English: Speaking skills, Feedback & questioning technique, Objectivity in argument (Both one on one and in groups), 5 Ws& 1 H & 7 Cs for effective Communication, Imbibing Etiquettes and manners, Study of different pictorial expressions of non-verbal communication and their analysis

UNIT II

Self-Management: Self-Management, Self Evaluation, Self-discipline, Self criticism, Recognition of one's own limits and deficiencies, dependency etc, Self Awareness, Self-Management, Identifying one's strengths and weaknesses, Planning & Goal setting, Managing self-emotions, ego, pride,- Leadership & Team Dynamics

UNIT III

Time Management Techniques: Practice by game playing and other learning strategies to achieve the set targets Time Management Concept, Attendance, and Discipline & Punctuality

UNIT IV

Motivation/ Inspiration : Ability to shape and direct working methods according to self-defined criteria, Ability to think for oneself, Apply oneself to a task independently with self-motivation, Motivation techniques: Motivation techniques based on needs and field situations

UNIT V

Interpersonal Skills Development: Positive Relationship, Positive Attitudes, and Empathies: comprehending others' opinions, points of views, and face them with understanding, Mutuality, Trust, Emotional Bonding, Handling Situations (Interview), Importance of interpersonal skills

Effective Computing Skills: Designing an effective Presentation: Contents, appearance, themes in a presentation, Tone and Language in a presentation, Role and Importance of different tools for effective presentation

Reference books:

1. Mitra, Barun, **"*Personality Development and Soft Skills"***, Oxford University Press, 2016.
2. Ramesh, Gopalswamy, **"*The Ace of Soft Skills: Attitude, Communication and Etiquette for Success"***, Pearson Education, 2013.
3. Covey, Stephen R., **"*Seven Habits of Highly Effective People: Powerful Lessons in Personal Change"***.
4. Osenberg Marshall B., **"*Nonviolent Communication: A Language of Life"***.

Course Title:	**Discrete Mathematics**	**Semester III**	
Course Code	**BTITC303**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To develop a foundation of set theory concepts, notation and applications.
2. To inculcate the habit of logical and mathematical thinking and its application to computer science and IT.
3. Understand logic, basic counting principles, relations, induction, sequences and summations.
4. To be able to present a coherent and mathematically accurate argument.
5. To understand the theory of graphs and algebraic structures and their applications.

Course Outcomes:
After learning the course the students should be able:

1. To perform operations on various discrete structures such as sets functions, relations and sequences.
2. To solve problems using counting techniques, permutation and combination, recursion and generating Functions.
3. To use graphs as tools to visualize and simplify problems.
4. To solve problems using algebraic structures (Rings, Monoids and Groups).

Course Content:

UNIT I

The Foundations: Sets theory and its applications sets, Set operations, Laws of set theory, Power sets, Partitions, Multi-sets, Cardinality, Principle of inclusion and exclusion, Algebra of sets and duality, Applications of sets: Problems on set operations and principle of inclusion-exclusion, Logics and proofs, Propositional logic, Propositional equivalences, Propositional algebra, Basic logical operations, De Morgan's laws, Predicates and quantifiers, Nested quantifiers, Rules of inference, Proof methods and strategy, Applications of logic: Translating English statements into propositions, Boolean searches in web pages, Bit operations.

UNIT II

Induction and recursion: Mathematical induction, Strong induction, Recursive definitions, Re-cursive algorithms, Applications: Proofs using mathematical induction, Program correctness, Well formed formula, Functions, Sequences and summations, Definition and types of functions: Injective, subjective and bijective , Composition, Identity and inverse of function, Re-cursively defined functions, Applications of functions, Job scheduling problem, Countability of rational numbers.

UNIT III

Basic Counting Principles: Permutations, Combinations, Binomial coefficients, Generalized permutations and combinations, Combinations and permutations with repetition, Generating permutations and combinations, Recurrence relation, Solving linear recurrence relations with constant coefficients, Applications of counting principles, Pigeonhole principle and its applications.

UNIT IV

Relations: Properties of binary relations, Closure of relations, Warshall's algorithm, Equivalence relations and partitions, Partial ordering relations and lattice application of relations: N-ary relations and their applications, Databases and relations. Algebraic Structures: Algebraic systems, Groups, Semi groups, Monoid, Subgroups, Permutation groups, Codes and group codes, Isomorphism and automorphisms, Homomorphism, Fermat's little theorem, Polynomial rings, Applications of groups, Ring, Field.

UNIT V

Graph Theory: Basic terminology, Multi graphs and weighted graphs, Paths and circuits, Shortest path in weighted graph, Hamiltonian and Euler paths and circuits, Factors of a graph, Shortest path algorithm, Traveling salesman problem. planar graph and Kuratowskis graph and theorem, independent sets, graph coloring. Trees, rooted trees, path length in rooted trees, binary search trees, spanning trees and cut set, theorems on spanning trees, cut sets , circuits, minimal spanning trees, Kruskal's and Prim's algorithms for minimal spanning tree.

Text Books:

1. K. H. Rosen, ***"Discrete Mathematics and Its Applications"***, Tata McGraw Hill Publication, 8th Edition, 2018.
2. J. P. Tremblay, R. Manohar, ***"Discrete Mathematical Structures with Applications to Computer Science"***, 1st Edition, McGraw Hill Publication, 2001.
3. C. L. Liu, "***Elements of Discrete Mathematics"***, McGraw-Hill Publication, 3rd Edition, 2008.

Reference Books:

1. B. Kolman, R. Busby, S. Ross, ***"Discrete Mathematical Structures"***, Pearson Education, 6th Edition, 2009.
2. R. K. Bisht, H. S. Dhami, ***"Discrete Mathematics"***, Oxford University Press, 2015.

Course Title:	**Object Oriented Paradigm with C++**	**Semester III**	
Course Code	**BTITC304**	**Course Type**	**Compulsory**
Prerequisite	**Programming in C**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. This course focuses on principles of object oriented programming paradigm.
2. This course includes practice of writing programs in C++.

Course Outcomes:

After learning the course, the students should be able:

1. To draw the control flow of a program.
2. To understand the storage concepts in a simple program.
3. To program using basic concepts of OO languages i.e. objects, encapsulation, data hiding etc.
4. To program using advanced concepts of OO languages i.e. associations, packages, interfaces, exception handling etc.
5. To work with functional, Logic programming paradigms.

Course Content:

UNIT I

Elements of computer systems, DOS commands and Linux environment, Algorithms, Flowcharts, Object-Oriented Programming Paradigm: Benefits, Applications, Object-Oriented Systems Development, Object-Oriented Analysis: Static and dynamic modeling, Object-Oriented Design: Class design and algorithm.

UNIT II

Beginning with C++: Tokens, Data types, Operators, Expressions, and Control structures, Array, Functions, Structures and Unions, Class and Objects, specifying a class, Defining member functions, Private member functions, Static data and member functions, Arrays of objects, Friend functions.

UNIT III

Constructors and Destructors: Constructor, Parameterized constructors, Multiple constructors in a class, Copy constructors, Dynamic constructors, Destructor. Programming for class diagram and relationship.

Inheritance: Single inheritance, Multilevel inheritance, Multiple inheritance, Hierarchical inheritance, Hybrid inheritance, Virtual base classes, Abstract classes.

UNIT IV

Polymorphism: Operator overloading, Function overloading, Virtual functions, pure virtual functions, Abstract class, Working with Files: Classes for file stream operations and I/O stream operation, Opening and closing a file, Detecting end-of-file, More about Open(): File Modes, Sequential input and output operations.

UNIT V

Exception Handling: Fundamentals, Types of exceptions, Catching exceptions, Multiple catching, Nested try statements, Uncaught exceptions, Throw and throws, Built-in exceptions, Creating exception subclasses, Using exceptions.

Text Books:

1. Robert Lafore, ***"Object Oriented Programming in C++"***, Pearson Education, 4th Edition, 2008.
2. E. Balagurusamy, ***"Object Oriented Programming with C++"***, Tata McGraw Hill Publication, 7th Edition, 2017.

Reference Books:

1. J. R. Hubbard, ***"Programming with C++: Schaum's Outlines"***, Tata McGraw-Hill publication, 3rd edition 2009.

2. P. J. Deitel, H.M.Deitel, ***"C++ How to Program"***, Pearson Education, 10th Edition, 2016.

Course Title:	**Data Structures and Applications**	**Semester III**	
Course Code	**BTITC305**	**Course Type**	**Compulsory**
Prerequisite	**Programming in C**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To assess how the choice of data structures and algorithm design methods affects the performance of programs.
2. To choose the appropriate data structure and algorithm design method for a specified application.
3. To solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, tournament trees, binary search trees, and graphs and writing programs for these solutions.

Course Outcomes:

After learning the course, the students should be able:

1. To write neat code by selecting appropriate data structure and demonstrate a working solution for a given problem.
2. To think of all possible inputs to an application and handle all possible errors properly.
3. To analyze clearly different possible solutions to a program and select the most efficient one.
4. To write an application requiring an effort of at least 1000 lines of code to demonstrate a good working solution.
5. To demonstrate the ability to write reusable code and abstract data types in C, using object-based way of thinking.

Course Content:

UNIT I

Introduction to Data Structures and Analysis of Algorithms: Need of data structures, Types of data structures, Recursion, ADT (Abstract Data Types), Basics of algorithm, Analysis of algorithm through time complexity and space complexity, Asymptotic notations.

UNIT II

Stack and Queue: Stack: Representation, Stack operation, Application. Queue: Representation, Queue operation, Circular and priority queue, Applications.

Linked list: Operation on linked list, Linked stacks and Queues, Array implementation of linked list, Linked list using dynamic variable, doubly linked list, Circular linked list.

UNIT III

Binary Tree: Basic tree concept, Binary tree operations, Binary tree representation, Binary tree traversals, Binary search tree and operations, Balanced tree: AVL trees and operations, Applications of binary trees, implementing priority queue using binary heap data structure.

UNIT IV

Graphs: Basics concepts of graphs, Representation of graphs, Graph traversals BFS and DFS, Minimum spanning tree algorithms: Kruskal's algorithm and Prim's algorithm, Applications of graphs.

UNIT V

Searching Techniques and Hashing: Linear search and binary search, Hashing: Direct-address tables, Hash tables, Open addressing, Perfect Hashing, Sorting techniques: Various sorting methods and their time complexity analysis: Insertion sort, Selection sort, Merge sort, Quick sort, Heap sort.

Text Books:

1. E. Horowitz, D. Mehta, S. Sahni, ***"Fundamentals of Data Structures in C++"***, Silicon Press, 2nd Edition, 2008.
2. Semour Lipschutz, ***"Data Structures with C"***, Tata McGraw-Hill, 1st Edition, 2010.
3. R.S. Bichkar, ***"Programming with C and Data structures"***, Universities Press, 1st Edition, 2014.

Reference Books:

1. Goodrich, Tamassia, "***Data Structures and Algorithm in Java***", Wiley publication, 6th Edition, 2014.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, "***Introduction to Algorithms***", MIT Press, 3rd Edition, 2009.
3. V. Goyal, L. Goyal, P. Kumar, "***A Simplified Approach to Data Structures***", Shroff Publishers, 1st Edition, 2014.

Course Title:	**Object Oriented Paradigm with C++ Lab**	**Semester III**	
Course Code	**BTITL306**	**Course Type**	**Compulsory**
Prerequisite	**Programming in C Lab**	**L – T – P**	**0 – 0 – 4**
Stream	**Core**	**Credits**	**2**

Lab Experiments List:

1. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power () that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main () function that gets values from the user to test this function.
2. A point on the two-dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:

   Enter coordinates for P1: 3 4
   Enter coordinates for P2: 5 7
   Coordinates of P1 + P2 are: 8, 11Create the equivalent of a four-function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally, it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be Y or N. Some sample interaction with the program might look like this:

   Enter first number, operator, second number: 10/ 3
   Answer = 3.333333
   Do another (Y/ N)? Y
   Enter first number, operator, second number 12 + 100
   Answer = 112
   Do another (Y/ N)? N
3. A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interaction might look like this:

   Enter your area code, exchange, and number: 415 555 1212
   My number is (212) 767-8900
   Your number is (415) 555-1212
   Create two classes DM and DB which store the value of distances. DM stores distances in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend functionto carry out the addition operation. The object that stores the results maybe a DMobject or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display.
4. Create a class rational which represents a numerical value by two double values- NUMERATOR and DENOMINATOR. Include the following public member Functions: constructorwith no arguments

(de-fault), constructor with two arguments, void reduce () thatreduces the rational number by eliminating the highest common factor between the numeratorand denominator.
Overload + operator to add two rational numbers
Overload - operator to enable input through cin
Overload * operator to enable output through cout
Write a main ( ) to test all the functions in the class.

5. Consider the following class definition:

```
class father {
protected age;
public;
father (int x) {age = x;}
virtual void iam()
{
cout<<“I AM THE FATHER ” ;
cout<< “My age is : ” <<age<<endl;}
};
```

Derive the two classes son and daughter from the above class and for each, define iam () to write similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main() that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam () through the pointer to demonstrate polymorphism in action.

6. Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll number, name (a string of 30 or less number of characters) and marks.

7. A hospital wants to create a database regarding its indoor patients. The information to store include:
Name of the patient
Date of admission
Disease
Date of discharge
Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the patients to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

8. Imagine a tollbooth with a class called tollbooth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function called nopayCar( ), increments the car total but adds nothing to the cash total. Finally, a member function called display() displays the two totals i.e. total cars and total cash. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

Course Title:	**Data Structures and Applications Lab**	**Semester III**	
Course Code	**BTITL306**	**Course Type**	**Compulsory**
Prerequisite	**Programming in C**	**L – T – P**	**0 – 0 – 4**
Stream	**Core**	**Credits**	**2**

Lab Experiments List:

Write a program:

1. To implement a character stack data type and use it to reverse a string.
2. To implement an integer stack data type that grows on demand.
3. To write a program using appropriate stacks for evaluating an infix expression with parenthesis.
4. To write a program, using a queue data type, to simulate a bank where customers are served on a first come first-serve basis.
5. To write one program for each of the following operations with singly linked lists:
   a. Concatenate two linked list and create third one
   b. Free all nodes in a linked list
   c. Reverse a linked list
   d. Given two linked list, create a third list which is set-intersection of the elements in the two.
6. To delete every third element from the linked list.
7. To copy a given linked list into another (new) list.
8. To implement a queue using a doubly linked list.
9. To write the following recursive functions for a singly-linked NULL-terminated list: insert(), traverse(), search().
10. To Implement the following sorting algorithms: (a) Insertion sort (b) Merge sort (c) Quick sort (d) Heap sort.
11. To implement a binary search tree (BST) by considering the keys in given order.
12. To implementation of graph traversal methods (a) BFS (b) DFS.

Course Title:	**Probability and Statistics**	**Semester IV**	
Course Code	**BTITC401**	**Course Type**	**Compulsory**
Pre-requisite	**Engineering Mathematics III**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. Be through with probability concepts.
2. To acquire knowledge on Probability Distributions.
3. Get exposed to the testing of hypothesis using distributions.
4. Gain strong knowledge inn principles of Queuing theory.
5. Get exposed to Discrete time Markov chain.

Course Outcome:

1. To acquire analytical ability in solving mathematical problems as applied to the respective branches of engineering.

Course Content:

UNIT I

Probability Theory: Definition of probability: classical, empirical and axiomatic approach of probability, Addition theorem of probability, Multiplication theorem of probability, Bayes' theorem of inverse probability, Properties of probabilities with proofs, Examples.

Random Variable and Mathematical Expectation: Random variables, Probability distributions, Probability mass function, Probability density function, Mathematical expectation, Join and marginal probability distributions, Properties of expectation and variance with proofs.

UNIT II

Theoretical Probability Distributions: Binomial distribution, Poisson distribution, Normal distribution, Fitting of binomial distributions, Properties of binomial, Poisson and normal distributions, Relation between binomial and normal distributions, Relation between Poisson and normal distributions, Importance of normal distribution, Examples.

UNIT III

Correlation: Introduction, Types of correlation, Correlation and causation, Methods of studying correlation, Karl Pearson's correlation coefficient, Spearman's rank correlation, Coefficient, Properties of Karl Pearson's correlation coefficient and Spearman's rank correlation coefficient, Probable errors. Linear Regression Analysis: Introduction, Linear and non-linear regression, Lines of regression, Derivation of regression lines of y on x and x on y, Angle between the regression lines, Coefficients of regression, Theorems on regression coefficient, Properties of regression coefficient.

UNIT IV

Testing of Hypothesis: Introduction to Sampling Distributions, Population and Sample, Null Hypothesis and Alternative Hypothesis, Single and Two Tailed Test. Testing of Hypothesis, Level of Significance, Critical Region, Procedure for Testing of Hypothesis Large Sample Test- Test for Single Proportion, Two Sample Proportions. Large Sample Test- Test for Single Mean, Two Sample Means. Small Sample Tests – "t" Test For a Single Mean "t" Test For The Difference Of Means, Paired "t" Test F Test – Test of Significance of the Difference between Two Population Variances. Chi Square Test for Goodness of Fit, Independence of Attributes.

UNIT V

Applied Statistics: Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves. Markov Chains: Introduction to Stochastic process, Markov process, Markov chain one step & n-step Transition Probability, Classification of states of a Markov chain – Applications.

Text Books:

1. Veerarajan T., ***"Probability, Statistics and Random Processes"***, Tata McGraw Hill, 1st Reprint 2004.
2. S.C. Gupta and V.K. Kapoor, ***"Fundamentals of Mathematical Statistics"***, Sultan Chand & Sons, 9th extensively revised Eition,1999.
3. G. V. Kumbhojkar, Probability and Random Processes, C. Jamnadas and Co., 14th Edition, 2010. 3. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
4. Veerarajan T., Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010.
5. G. Haribaskaran, Probability, Queuing Theory and Reliability Engineering, Laxmi Publications, 2nd Edition, 2009.
6. Murray Spiegel, John Schiller, R. ALU Srinivasan, Probability And Statistics, Schaum's Outlines, 4th Edition, 2013.

Reference Books:

1. Trivedi K S, ***" Probability and Statistics with reliability, Queueing and Computer Science Applications***", Prentice Hall of India, New Delhi, 1984
2. Gross.D, Harris.C.M. , ***"Fundamentals of Queuing Theory"***, John Wiley and Sons, 1985.
3. Allen.A.O.,***"Probability Statistics and Queuing Theory"***, Academic Press, 1981

Course Title	**Computer Architecture and Organization**	**Semester IV**	
Course Code	**BTITC402**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To understand the structure, functions and characteristics of computer systems.
2. To learn basics of Computer Architecture.
3. To study hierarchical memory system including cache memories and virtual memory.
4. To identify input / output devices and their data transfer mechanisms.

Course Outcomes:

After learning the course, the students should be able:

1. To identify components of a computer system including CPU, memory and input/output units.
2. To explain instruction types, its execution and interrupt mechanism.
3. To illustrate numerical and character representations in digital logic and floating point arithmetic.

Course Content:

UNIT I

Introduction: Concept of computer organization and architecture, Fundamental unit, Computer function and interconnection, CPU structure and function.

UNIT II

Instruction Sets: Characteristics, Types of operands, Types of operations, Assembly language, Addressing modes, Instruction format, Types of instruction, Instruction execution, Machine state and processor status, Structure of program, Introduction to RISC and CISC architecture.

Computer Arithmetic: The arithmetic and logic Unit, Integer representation, Integer arithmetic, Floating point representation, Floating point arithmetic, Introduction of arithmetic co-processor.

UNIT III

Memory Organization: Internal Memory: Semiconductor main memory, Error correction, Advanced DRAM organization, Virtual memory systems and cache memory systems, External Memory: Organization and characteristics of magnetic disk, Magnetic tape, Optical memory, RAID, Memory controllers.

UNIT IV

Control Unit: Control unit operation: Micro-operations, Control of the processor, Hardwired implementation, Micro-programmed Control Unit, Basic concepts, Micro-instruction sequencing, Micro-instruction execution, Applications of micro-programming.

UNIT V

Input/ Output Organization**:** External devices, I/O module, Programmed I/O, Interrupt driven I/O, Direct memory access, I/O channels and processors, External interface. Instruction pipe-lining: Concepts, Parallel processing: Multiple processor organization, Symmetric multiprocessor, Cache coherence and the MESI protocol.

Text Books:

1. William Stalling, ***"Computer Organization and Architecture: Designing for Performance"***, 8th Edition, Prentice Hall Publication, 2009.
2. Hayes, ***"Computer Architecture and Organization"***, 3rd Edition, McGraw-Hill Publication, 2012.
3. Zaky, **"*Computer Organization"***, 5th Edition, McGraw-Hill Publication, 2011.

Reference Books:

1. Morgan and Hennessy and Patterson, ***"Computer Architecture: A Quantitative Approach"***, 4th Edition, Kaufman Publication, 2007.
2. Morris Mano, ***"Computer System Architecture"***, 3rd Edition, Pearson Education India, 2007.
3. MostafaAbd-El-Barr, Hesham El-Rewini, ***"Fundamentals of Computer Organization and Architecture"***, 1st Edition, Wiley Publication, 2004.
4. Miles J. Murdocca, Vincent P. Heuring, ***"Computer Architecture and Organization: An Integrated Approach"***, 1st Edition, Wiley Publication, 2007.

Course Title:	**Design and Analysis of Algorithms**	**Semester IV**	
Course Code	**BTITC403**	**Course Type**	**Mandatory**
Pre-requisite	**Data Structures and Applications**	**L – T – P**	**3 – 1– 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To learn fundamentals of algorithms design techniques.
2. To understand basic knowledge of computational complexity, approximation and randomized algorithms, selection of the best algorithm to solve a problem.
3. To analyze the performance of algorithms, to compare algorithms with respect to time and space complexity.
4. To develop proficiency in problem solving and programming.

Course Outcomes:

After learning the course the students should be able:

1. Develop efficient algorithms for simple computational tasks.
2. Gain understanding of concepts of time and space complexity, worst case, average case and best case complexities and the big-O notation.
3. Design standard algorithms such as sorting, searching, and problems involving graphs.
4. Compute complexity measures of algorithms, including recursive algorithms using recurrence relations.

Course Content:

UNIT I

Introduction: Instruction counts, Growth functions, Necessity of time and space analysis of algorithms, Order notations (O, Θ, Ω notations), Problem instance size, frequently occurring recurrence relations in analysis of algorithms.

UNIT II

Design Techniques-I: Divide and Conquer: Binary search, finding maximum and minimum, Merge sort, Quick sort, Strassen's matrix multiplication. Greedy Algorithms: Knapsack problem, Job sequencing with deadlines, optimal storage on tapes, Optimal merge pattern, Single source shortest paths.

UNIT III

Design Techniques-II: Dynamic Programming: Multistage graphs, All pairs shortest paths, 0/1 Knapsack, Travelling salesman problem.

Design Techniques-III: Backtracking: 8-Queens Problems, Sum of subsets, Graph coloring. Branch-and-bound: Least cost (LC) search, Control abstractions for LC search, FIFO branch and bound, LC branch and bound.

UNIT IV

Selected Algorithms from Various Areas: Graph Theory, Elementary Algorithms: DFS, BFS, Topological Sort, Minimum spanning trees (Kruskal and Prim's algorithms), Shortest Paths: Single source shortest paths, all pairs shortest paths, String Matching: The naive string-matching algorithm, The Robin-Karp algorithm, The Knuth-Morris-Pratt algorithm.

UNIT V

Complexity Theory: Lower-bound arguments, NP-completeness: Introduction to NP-Complete, Reducibility (SAT, Independent Set, 3VC, Subset Sum and Partition, Hamiltonian Circuit).

Text Books:

1. Thomas Cormen, Charles Leiserson, Ronald Rivest and Cliford Stein, ***"Introduction to Algorithms"***, MIT Press, 3rd Edition, 2010.
2. E. Horowitz, S. Sahni and S. Rajsekaran, ***"Computer Algorithms"***, Silicon Press, 2nd Edition, 2008.

Reference Books:

1. B. K. Joshi, ***"Data Structures and Algorithms in C++"***, Tata McGraw Hill Education, 2010.
2. G. T. Heineman, Gary Pollice, Stanley Selkow, ***"Algorithms in a Nutshell"***, Shroff Publication, 2nd Edition, 2016.
3. Kyle Loudon, ***"Mastering Algorithms with C"***, OReilly Publication, 1st Edition, 2009.

Course Title:	**Organizational Behavior**	**Semester IV**	
Course Code	**BTITC404**	**Course Type**	**Compulsory**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Humanities, Social Science and Management**	**Credits**	**3**

Course Objective:

1. To explore the organization as a micro-social system - a medium to facilitate and improve the interpersonal relationships in the context of organizational functioning.

Course Outcomes:

1. Students will become more self-aware and will have identified areas of development for long term effectiveness.
2. Students will understand the role that individuals play collectively to perform in organizations.

Course Content:

UNIT I

Introduction to Organizational Behavior: Management and organizational behavior, Complementing intuition with systematic study, Disciplines and opportunities for OB, developing an OB Model. diversity in organizations: Diversity, Discrimination, biographical characteristics, other differentiating characteristics, ability, implementing diverging managements strategies.

UNIT II

Attitudes and job satisfaction, Attitudes, Attitudes and behavior, Measuring job satisfaction, what causes job satisfaction, Outcomes of job satisfaction, the impact of job dissatisfaction, emotions and moods, sources of emotions and moods, emotional labor, affective events theory, emotional intelligence, emotion regulation, OB application of emotions and moods, personality and values, personality, personality frameworks, other personality attributes relevant to OB, personality and situation, values, linking an individual's personality and values to the workplace, cultural values.

UNIT III

Perception and individual decision, what is person perception?, perception: Making judgments about others, the link between perception and individual decision making, decision making in organizations, influences on decision making: individual differences and organizational constraints, what about ethics in decision making?, Creativity: creative decision making, and innovation in organizations, motivation concepts, early theories of motivation, contemporary theories of motivation, other contemporary theories of motivation, jab engagement, integrating contemporary theories of motivation, motivation: from concepts to application, motivating by job design: the Job Characteristic Model (JCM) using job redesign to motivate employees using alternative work arrangements to motivate employees, using Employees Involvements and Participation (EIP) to motivate employees, using extrinsic rewards to motivate employees, using benefits to motivate employees, using intrinsic. Rewards to motivate employees, motivation employees through company culture.

UNIT IV

Foundation of group behavior, group and group identity, Stages of group development, group, differences between group and teams, types of teams, creating effective teams, turning individuals into team players, communication, Direction of communication, modes of communication, choice of communication, persuasive communication, cultural factors, rewards, competency based rewards, performance based rewards, Empowerment and self-managed teams.

UNIT V

Leadership, trait theories of leadership, behavioral theories, contingency theories, contemporary theories of leadership, responsible leadership positive leadership, challenges of leadership, power and leadership bases of power, Power tactics, how power affects people, politics: power in action, causes and consequences of political behavior, A definition of conflict, the conflict process, negotiation, the negotiation process, third party negotiation. Foundations of organization structure, common organizational frameworks structures, alternate design options, organizational designs and employees behavior, organizational culture, creating and sustaining culture, , organizational change and stress management, resistance to change, approaches to managing organizational change

.

Text books:

1. Uma Sekaran, **"*Organization Behaviors"***, McGraw Hill Company, New Delhi, 2011.
2. LM Prasad, ***"Organization Behavior"***, S. Chand and Co. Ltd, New Delhi, 2008.
3. Nair, Banerjee, Agarwal, **"*Organization Behavior"***, Pragati Prakashan, New Delhi, 2006.
4. Stephen P. Robbins, Timothy A. Judge, Neharika Vohra, ***"Organizational Behaviour"***, Pearson, 14th edition,2014.

Reference books:

1. Rosy Joshi and Sashi K Gupta, **"*Organization Behaviors"***. Kalyani publishers, New Delhi, 2005.
2. S.S. Khanka, **"*Organization Behavior"***, S. Chand and Co. Ltd, New Delhi, 2008.
3. Fred Luthans, **"*Organizational Behavior"*,** McGraw Hill Book Co., 2005.

Course Title:	**Microprocessors and Microcontrollers**	**Semester IV**	
Course Code	**BTITPE405A**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**2– 1 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To understand 8086 microprocessor Architecture.
2. To understand design aspects of I/O and Memory Interfacing circuits.
3. To acquaint with instruction set and logic required to build assembly language programs.
4. To learn micro-controller architecture, its instruction set and interfaces.

Course Outcomes:

After learning the course the students should be able:

1. To design and implement programs on 8086 microprocessor.
2. To design I/O circuits and Memory Interfacing circuits.
3. To exhibit knowhow on micro-controller interfaces & programming.
4. To experiment with MCS51 and PIC18 micro-controller.

Course Content:

UNIT I

Intel 8086/8088 Microprocessor Family: Architecture and organization of 8086/8088 microprocessor family, Instruction set, Assembly language programming, Introduction to mixed language programming using C and Assembly language, 8086 family minimum and maximum mode operation, Timing diagram for 8086 family, Detailed study of maximum mode connection: Study of 8288 bus controller, 8086 interrupt structure.

UNIT II

8086 Instruction Set and Programming: Addressing modes, Instruction Set, ALP, Mixed language programming, Stacks, Strings, Procedures, Macros, Timers, Counters and delay, Programming examples using DOS and BIOS Interrupts, Device drivers programming.

8086 Interrupt System: 8086 Interrupt structure, Types and applications: Study of Interrupt Controller 8259A and Interrupt Priority Management using 8259A.

UNIT III

Memory System Design and I/O Interfacing: Interfacing SRAM, ROM and DRAM to 8086, Address decoding and Timing Considerations, I/O interfacing in 8086: Serial communication interface includes Synchronous and Asynchronous, Protocols, Parallel communication interface includes I/O Mapped I/O, Memory Mapped I/O, and Handshaking Signals, 8087 Math Co-processor: Study of architecture of 8087, Floating point coprocessor, Data types supported by 8087, Host and coprocessor interface, Assembly language Programming for 8086 - 8087 based systems.

UNIT IV

Intel MCS 51 Family: Introduction to Single chip microcontrollers of Intel MCS 51 family, Architectural and operational features, Instruction set, CPU timing and machine cycles, Interrupt structure and priorities, Internal Timer / counters, Serial interface, Connection of external memory, Power saving modes, Interfacing of 8051 with EPROM, Programming for EPROM versions, 8051 variation.

UNIT V

Introduction to the PIC18 Microcontroller: Overview of the PIC18 MCU, The PIC18 Memory Organization, The PIC18 CPU Register, The PIC18 Pipelining, PIC18 Instruction Format, Addressing Modes, A Sample of PIC18 Instruction, Overview of the 8-Bit MCU Market.

Text Books:

1. Douglas Hall, ***"Microprocessors and Interfacing: Programming and Hardware"***, Tata McGraw-Hill, 2nd Edition.
2. Han-Way Huan, ***"An Introduction to Software and Hardware Interfacing"***, Delmar Cengage Learning, 2nd Edition, 2006.

Reference Books:

1. Peter Norton, ***"IBM PC, Assembly Language programming"***, BPB publication.
2. John Uffenback, ***"8086/8088 Interfacing, Programming and Design"***, Prentice Hall of India Publication.
3. A. K. Ray, K. M. Bhurchandi, ***"Advanced Microprocessors and Peripherals"***, Tata McGraw Hill, 2000.
4. John Uffenback***, "8086/8088 Interfacing, Programming and Design",*** Prentice Hall of India Publication.

Course Title:	**Web Technology**	**Semester IV**	
Course Code	**BTITPE405B**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**2 – 1– 0**
Stream	**Professional Core**	**Credits**	**3**

Course Objectives:

1. Overview of modern Web technologies.
2. To use different web scripting technology.
3. To understand web hosting, server type, debugging, and performance driven application development.
4. To understand user interface and awareness of real-world knowledge.

Course Outcomes:

1. To understand World Wide Web and latest trends in web-development.
2. Real world knowledge of design and development.
3. Design and development of web application with all industrial standards.
4. Awareness of web hosting, server type, debugging.

UNIT I

Introduction to World Wide Web, Features of web, HTTP, Web Servers, Introduction to Scripting Language, Browser, Integrated Development Environment.

UNIT II

HTML: Introduction to HTML, Basics of HTML, Formatting and fonts, Commenting code, HTML heading, Block element, Inline element, Comment, Attributes, Hyperlink, Lists, Tables, Images, Forms, Meta tags, Character entities, Frames sets.

UNIT III

Advance HTML: Overview and features of HTML5, Includes External File, Responsive Layout with Media Queries, Marquee, Semantic Tags, HTML Symbol, URL Encode, Cashing, Video Tags, Audio Tags, Image Maps. CSS: Introduction To CSS, Selector, Basic Syntax And Structure, Padding, Margin, Manipulating Texts, Display, Height, Width, Border, Color, Fonts, Positioning Using CSS, Overview And Features Of CSS3.

UNIT IV

PHP**:** Introduction to PHP, Features of PHP, Basics of PHP, Syntax, Variable, Printing Output, Array, String, Function, Data types, Operator, Loops, Conditional Statement, Introduction To Advance PHP, Form Processing, Files, PHP Cookies, PHP Sessions, Constant, PHP Magic Function, PHP Global Variable, Error Handling, Exception, Connection with Database, Curd Operation in PHP.

UNIT V

Web Hosting, Debugging and Unit Testing, Browser Compatibility.

Text Book:

1. Snehal Joglekar, **"*HTML and CSS- Web Technologies*"**, Nirali Prakashan, 2016.

Reference Books:

1. Thomas Powell, **"*HTML & CSS: The Complete Reference*"**, 5th Edition, McGraw Hill Publication.
2. Steven Holzner, **"*PHP: The Complete Reference*"**, 1st Edition, McGraw Hill Publication.

Course Title:	**Physics of Engineering Materials**	**Semester IV**	
Course Code	**BTITPE405C**	**Course Type**	**Elective**
Prerequisite	**Engineering Physics**	**L – T – P**	**2 – 1– 0**
Stream	**Professional Core**	**Credits**	**3**

Course Objectives:
To understand and apply the Physics principles behind the development of Engineering Materials.

Expected Outcome:
The students will be able to understand fundamentals of Electrodynamics, Crystal structure, Semiconductors, Dielectrics, Nano materials, Magnetic and superconducting materials. It forms the base of many modern advance devices and technology.

UNIT I

Crystallography : Crystal direction and planes, Diatomic Crystal (CsCl, NaCl, Diamond, $BaTiO_3$) Crystal imperfection, point defects, line defects, Surface and volume defects, Structure properties relationship, structure determination by X-ray diffraction.

UNIT II

Magnetic Materials : Origin of magnetization using atomic theory classification of magnetic materials and properties, Langevin's theory and Dia, Para and ferromagnetism, Soft and Hard Magnetic materials and their uses, Domain theory of ferromagnetism, Hysteresis loss, Anti-ferromagnetic and Ferromagnetic materials, Ferrites and Garnets, magnetic bubbles, magnetic recording.

UNIT III

Conducting and Superconducting Materials : Band theory of solids, Classical free electron theory of metals, Quantum free electron theory, Density of energy and carrier concentration, Fermi energy, Temperature and Fermi energy distribution, Superconductivity, Factor affecting Superconductivity, Meissner effect, Type-I and Type-II superconductors, BCS theory, Josephson effect, High temperature superconductor, Application of superconductors (Cryotron, magnetic levitation)

UNIT IV

Semiconducting Material : Band structure of semiconductor, Charge carrier concentration, Fermi level and temperature, Electrical conductivity, Hall effect in semiconductor, P-N junction diode, preparation of single crystals, LED, photovoltaic Cell

UNIT V

Dielectric Materials : Dielectric constant and polarizability, types of polarization, temperature and frequency dependences of Dielectric parameter, internal fields in solids, Clausius - Mosotti equation, dielectric loss, dielectric breakdown, ferroelectric, pyroelectric and piezoelectric materials, applications of dielectric materials.
Nanomaterials: Introduction and properties, synthesis of nanomaterials, Carbon Nano Tubes, Characterization techniques of nano materials- SEM, TEM, EDAX, FMR, XRD. Applications of nanomaterials.

Text Book

1. C. Kittle, ***"Introduction to Solid state Physics"*** .
2. C.M. Srivastava and C. Srinivasan, ***"Science of Engineering Materials and Carbon Nanotubes"***.
3. A.J. Dekker, ***"Solid State Physics".***

Reference Books

1. V raghavan, ***"Material Science and Engineering".***
2. A.J. Dekker, ***"Electrical Engineering Material" .***

Course Title:	**Soft Skills and Personality Development**	**Semester IV**	
Course Code	**BTCOE406C**	**Course Type**	**Compulsory**
Pre-requisite	**Nil**	**L – T – P**	**2 – 0 – 0**
Stream	**Nil**	**Credits**	**2**

Course Objectives:

1. To build the skills like team building so that they can work efficiently in groups.
2. To provide knowledge of conflict management while working in large organisations.
3. To develop management skills required in routine work environment.
4. To polish the personality of the learners in order to make them good leaders and employees.
5. To imbibe qualities like manners & etiquettes co-ordination, mutual understanding for their colleagues while working in a group.

Course Outcomes:

1. Learners will acquire interpersonal communication skills.
2. Learners will develop the ability to work independently.
3. Learners will develop the qualities like self-discipline, self-criticism and self-management.
4. Learners will have the qualities of time management and discipline.
5. Learners would be able to present themselves as an inspiration for others.

UNIT I

Introduction to Soft Skills and Personality Development Introduction to Soft Skills, Need of Soft Skills, New Approach to Learning Human Perceptions: Understanding People Types of Soft Skills: Self-Management Skills, Interpersonal Skills, What is Personality, Personality Development

UNIT II

Self Management & Self Management Techniques: Self Management, Stress Management: Types of Stress: Self-Awareness about Stress, Regulating Stress: Making The Best out of Stress, Self-Evaluation, Self- discipline, Self-criticism, Recognition of one's own limits and deficiencies, dependency etc. Self-Awareness, Self-Management, Identifying one's strengths and weaknesses, Planning & Goal setting, Managing self-emotions, ego, pride.

UNIT III

Interpersonal Skills Development: Positive Relationship, Positive Attitudes, Empathies: comprehending others' opinions, points of views, and face them with understanding, Mutuality, Trust, Emotional Bonding, Handling Situations (Interview), Importance of interpersonal skills, Creative Thinking, Critical Thinking

UNIT IV

Conflict Resolution Skills: Seeking Win-Win Solution (Negotiation Skills), Inter-Personal Conflicts: Two Examples, Inter-Personal Conflicts: Two Solutions, Types of Conflicts: Becoming a Conflict Resolution Expert

UNIT V

Motivation/ Inspiration: Ability to shape and direct working methods according to self-defined criteria, Ability to think for oneself, Apply oneself to a task independently with self-motivation, Motivation techniques, Motivation techniques based on needs and field situations, Leadership and Team Dynamics

Reference books:

1. Mitra, Barun, "***Personality Development and Soft Skills***", Oxford University Press, 2016
2. Ramesh, Gopalswamy, *The Ace of Soft Skills: Attitude, Communication and Etiquette for Success,* Pearson Education, 2013
3. Covey, Stephen R., ***"Seven Habits of Highly Effective People: Powerful Lessons in Personal Change*"**
4. Rosenberg Marshall B., ***"Nonviolent Communication: A Language of Life"***

Course Title:	**Design and Analysis of Algorithms Lab**	**Semester IV**	
Course Code	**BTITL406**	**Course Type**	**Mandatory**
Pre-requisite	**Data Structures and Applications Lab**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments List:

1. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n and record the time taken to sort. Plot a graph of the time taken versus non graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate how the divide and conquer method works along with its time complexity analysis: worst case, average case and best case.
2. Implement the 0/1 Knapsack problem using (a) Dynamic Programming method (b) Greedy method.
3. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm. Write the program.
4. Find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm. Use Union-Find algorithms in your program.
5. Find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.
6. Write programs to (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm
   (b) Implement Travelling Sales Person problem using Dynamic programming.
7. Design and implement a program to find a subset of a given set S = Sl, S2,.....,Sn of n positive integers whose SUM is equal to a given positive integer d. For example, if S = 1, 2, 5, 6, 8 and d = 9, there are two solutions 1, 2,6and 1, 8. Display a suitable message, if the given problem instance doesn't have a solution.
8. Design and implement a program to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.

Course Title:	**Microprocessors and Microcontrollers Lab**	**Semester IV**	
Course Code	**BTITEL406**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments List:

I. **Assembly Language Program – 8086 using TASM:**

1. Programs on Arithmetic and Logical Instructions of 8086
   i) Addition of two 16-bit numbers
   ii) Addition of two multi byte numbers
   iii) Product of two multi byte numbers
   iv) Division of two multi byte numbers
2. Factorial of a number
3. Move a String from one location to other location
4. Display a String in reverse order
5. Compute length of a given String
6. Insert a character at the end of the given string
7. Delete first character of a given string
8. Programs on Data Transfer Instructions of 8086
9. Find sum of ASCII numbers
10. Programs on String instructions of 8086
11. Programs on Subroutines of 8086
12. Convert two-digit Packed BCD to Unpacked BCD
13. DOS/BIOS Programming, reading keyboard (buffered with and without echo) -Display characters
14. Sorting of an Array

II. **Microcontroller 8051:**

1. Reading and Writing data on a parallel port.
2. Timer in different modes.
3. Serial communication implementation.

III. **Interfacing with 8051:**

1. Programming using arithmetic, logical and bit manipulation instructions of 8051
2. (a) Program and verify Timer/Counter in 8051
   (b) Program and verify Interrupt handling in 8051.

Course Title:	**Web Technology Lab**	**Semester IV**	
Course Code	**BTITL406**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Professional Core**	**Credits**	**1**

Lab Experiments List:

1. Download XAMPP or WAMPP server, IDE, browsers to run HTML program.
2. Develop page to display fruits list with different color with heading on top of the page and link each fruit with fruit description page.
3. Develop using semantic element, page having menu bar in header section.
4. Develop user personal info form using HTML5 input control and decorate with CSS.
5. Develop responsive page layout using media queries.
6. Write a PHP program to print list of user info using array.
7. Write a PHP program to fetch user info from MYSQL database.
8. Write a PHP program to perform crud operation.
9. Write a PHP function to check palindrome string.
10. Write a PHP program using for loop to add all the integers between 0 and 30 and display the total.
11. Create a script to construct the pyramid of asterisk ('*') using nested for loop.
12. Write a program to calculate factorial of a number using for loop.
13. Write a program which will count the specific characters in the text
14. Debug web site using developer tools, inspect element.

Course Title:	**Physics of Engineering Materials Lab**	**Semester IV**	
Course Code	**BTITL406**	**Course Type**	**Elective**
Prerequisite		**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments List:

1. Crystallite size of nanomaterial's using XRD
2. B-H loop experiment.
3. Susceptibility measurement.
4. Resistivity of materials by four probe method.
5. Hall effect- determination of carrier concentration.
6. G.M. Counter : Study of Nuclear Counting Statistics
7. Dielectric constant measurement

Course Title:	**Software Engineering**	**Semester V**	
Course Code	**BTITC501**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To understand software life cycle development models.
2. To understand and apply software requirements engineering techniques, software design principles, modeling and software testing techniques.
3. To understand the use of metrics in software engineering.
4. To understand software project management.

Course Outcomes:

After learning the course the students should be able:

1. To use the techniques, skills, and modern engineering tools necessary for engineering practice.
2. To design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
3. To identify, formulate and solve engineering problems.

Course Content:

UNIT I

Software Development Process: Software crisis and myths, Software process and development: Generic view of process, Software life cycle and models, Analysis and comparison of varies models, an agile view of process, Requirement Engineering: Requirements engineering tasks, Initiating requirement engineering process, Eliciting requirement, developing use-cases.

UNIT II

Building the analysis model, Negotiating and validating requirement, Building the analysis model, System Design Overview: Design process and design quality, Design concepts, Design model, Pattern based software design, Architectural design, User interface design. UML: Different methods: Rumbaugh / Booch / Jacobson, Need for standardization. Developing diagrams in UML (Use CASE, Class, Interaction, State diagrams) CASE TOOLS.

UNIT III

Validation and Testing: Strategic approach to Software testing, Strategic issues, Test strategies for conventional software, Validation testing, System testing, Debugging. White box testing and Black box testing.

UNIT IV

Planning of Project: Project management, Metrics for process and projects, Estimation, Project scheduling.

UNIT V

Management of Project: Risk management, Importance of software quality and measurements software engineering techniques for quality assurance, and Change management. ISO 9000 and CMM/PCMM.

Text Books:

1. Roger S. Pressman, "***Software Engineering***", Tata McGraw-Hill, 6th Edition, 2006.
2. G. Booch, J. Rambaugh, and I. Jacobson, "***The Unified Modeling Language User Guide***", Addison Wesley, 2nd Edition, 2005.

Reference Books:

1. Shari Pfleeger, "***Software Engineering***", Pearson Education, 3rd Edition, 2008.
2. Ian Sommerville, "***Software Engineering***", Pearson Higher Education, 10th Edition, 2016.
3. Pankaj Jalote, "***An Integrated Approach to Software Engineering***", Springer New York, 2nd Edition, 2013.

Course Title:	**Computer Networks and Internetworking Protocols**	**Semester V**	
Course Code	**BTITC502**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To understand the basic concepts of Computer Networks.
2. To Understand Network Layer and Applications.
3. To learn UDP and TCP applications.
4. To learn Transport Layer Reliability.

Course Outcomes:

After learning the course, the students should be able:

1. To compare and contrast TCP and UDP in terms of the application that uses them.
2. To design network-based applications using the socket mechanism.
3. To work with IPv4 addresses in terms of subnetting and supernetting.
4. To setup a host and network in terms of IP addressing.

UNIT I

Introduction to Computer Networks and Underlying Technologies : ARPANET, Birth of the Internet, MILNET , CSNET , NSFNET, ANSNET, The Internet Today, World Wide Web, Time Line, Growth of the Internet, Protocols and Standards, Standards Organizations: Internet Standards Internet Administration.

The OSI Model and the TCP/IP Protocol Suite:

Protocol Layers: Hierarchy Services, The OSI Model: Layered Architecture , Layer-to-Layer Communication, Encapsulation, Layers in the OSI Model, TCP/IP Protocol Suite: Comparison between OSI and TCP/IP Protocol Suite, Layers in the TCP/IP Protocol Suite, Addressing: Physical Addresses, Logical Addresses, Port Addresses, Application-Specific Addresses, Wired Local Area Networks: IEEE Standards, Frame Format, Addressing, Ethernet Evolution, Standard Ethernet, Fast Ethernet, Gigabit Ethernet, Ten-Gigabit Ethernet.

UNIT II

Wireless LANS: IEEE, MAC Sublayer, Addressing Mechanism, Bluetooth, Point-to-Point WANs, DSL Technology, Cable Modem, ATM, Connecting devices: Repeaters, Bridges and Routers.

Introduction to Network Layer: Switching: Packet Switching, Circuit Switching, Packet Switching at Network Layer, Network Layer Services, Other Network Layer Issues. IPv4 Addresses, Address Space Notation, Range of Addresses, Operations, Classful Addressing: Classes, Classes And Blocks, Two-Level Addressing, Three-Level Addressing: Subnetting, Supernetting, Classless Addressing: Variable-Length Blocks, Two-Level Addressing, Block Allocation, Special Addresses: Special Blocks, Special Addresses in Each block, NAT, Address Translation, Translation Table.

UNIT III

Delivery and Forwarding of IP Packets: Direct Delivery, Indirect Delivery, Forwarding Based on Destination Address, Forwarding Based on Label, Structure of a Router: Components.

Internet Protocol Version 4(IPv4): Datagrams, Fragmentation, Maximum Transfer Unit (MTU), Fields Related to Fragmentation, Options: Format, Option Types, Checksum: Checksum Calculation at the Sender, Checksum Calculation at the Receiver, Checksum in the IP Packet, IP PACKAGE : Header-Adding Module, Processing Module, Queues, Routing Table, Forwarding Module, MTU Table, Fragmentation Module, Reassembly Table, Reassembly Module.

Address Resolution Protocol (ARP): Address Mapping: Static Mapping, Dynamic Mapping, The ARP Protocol: Packet Format, Encapsulation, Operation, Proxy ARP, ARP Package: Cache Table, Queues, Output Module, Input Module, Cache-Control Module.

Internet Control Message Protocol (ICMP): Messages: Message Format, Error Reporting Messages, Query Messages, Checksum, Debugging Tools: Ping, Traceroute, ICMP Package: Input Module, Output Module, Introduction to routing protocols.

UNIT IV

Introduction to Transport Layer: Transport-Layer Services: Process-to-Process communication, Addressing: Port Numbers, Encapsulation and Decapsulation , Multiplexing and Demultiplexing, Flow Control, Error Control , Combination of Flow and Error Control, Congestion Control, Connectionless and Connection-Oriented Services.

User Datagram Protocol (UDP): User Datagram, UDP Services: Process-to-Process Communication, Connectionless Services, Flow Control, Error Control, Congestion Control, Encapsulation and Decapsulation, Queuing, Multiplexing and Demultiplexing, Comparison between UDP and Generic Simple Protocol, UDP Applications: UDP Features, Typical Applications, UDP Package: Control-Block Table, Input Queues, Control-Block Module, Input Module, Output Module.

UNIT V

Transmission Control Protocol (TCP): TCP Services: Process-to-Process Communication, Stream Delivery Service, Full-Duplex Communication, Multiplexing and Demultiplexing, Connection-Oriented Service, Reliable Service. TCP Features: Numbering System, Flow Control, Error Control, Congestion Control, Segment: Format, Encapsulation, A TCP Connection: Connection Establishment, Data Transfer, Connection Termination, Connection Reset, State Transition Diagram, Scenarios ,Windows in TCP ,Send Window, Receive Window, Flow Control : Opening and Closing Windows, Shrinking of Windows, Silly Window Syndrome, Error Control :Checksum, Acknowledgment, Retransmission, Out of-

Order Segments, Data Transfer in TCP, Some Scenarios, Congestion Control : Congestion Window, Congestion Policy, TCP Timers: Retransmission Timer, Persistence Timer, Keepalive Timer, TimeWait Timer, TCP Package: Transmission Control Blocks (TCBs), Timers, Main Module, Input Processing Module, Output Processing Module.

Text books:

1. Douglas E. Comer, "***Internetworking with TCP/IP: Principles, Protocols and Architecture***", Volume 1, 6th Edition, PHI publication, 2013.
2. Behrouz A. Forouzan, "***TCP-IP Protocol Suite***", 4th Edition, McGraw Hill publication, 2010.

Reference books:

1. Comer, "***Internetworking with TCP-IP***", Volume 3, 5th Edition, Pearson publication, 2013.
2. W. Richard Stevens, "***UNIX Network Programming: Interprocess Communications***", Volume 2, 2nd Edition, PHI publication, 1999.
3. William Stalling, "***SNMP, SNMPv2, SNMPv3, and RMON 1 and 2***", 2nd Edition, Pearson education publication, 2001.
4. Hunt Craig, "***TCP-IP Network Administration***", 3rd Edition, O'Reilly publication, 2002.
5. Loshin, Harwurt, "***TCP-IP Cleanly Explained***", BPB publication.

Course Title:	**Embedded Systems**	**Semester V**	
Course Code	**BTITPE503A**	**Course Type**	**Elective**
Pre-requisite	**Microprocessor & Microcontroller**	**L – T – P**	**3 – 0 – 0**
Stream	**Software Application and Development**	**Credits**	**3**

Course Objectives:

1. To understand the fundamental concepts in Embedded Systems.
2. To learn Real Time Operating Systems.
3. To get acquainted with hardware & interfaces.
4. To know Embedded System Design Techniques.

Course Outcomes:

After learning the course the students should be able:

1. To demonstrate & explain embedded systems hardware & software components.
2. To define embedded systems using real time operating system – VxWorks/ μCOS II RTOS.
3. To design & develop embedded applications using C language.
4. To apply design techniques in real-life application.

Course Content:

UNIT I

Introduction: Introduction to embedded systems overview, design challenges, common design metrics, processor technology, IC technology, Design technology. Design productivity gap.

ARM Architecture: ARM 7 processor fundamentals, memory management, ARM processor family, Instruction set & interfacing. Introduction to ASIPS, Microcontrollers and DSP.

UNIT II

Devices and Interfacing: Processor interfacing, Arbitration, Multilevel bus architecture. Basic protocol concepts: serial protocols, I2C,CAN, Firewire and USB, Parallel protocols, PCI bus, ARM bus, Wireless protocols: IrDA, Bluetooth, IEEE 802.11, Device Driver programming.

UNIT III

Programming concepts: State m/c & concurrent process model, FSM m/c, FSMD, PSM model & concurrent process model, Scheduling process, Data flow model, Embedding programming in C++, JAVA and program modeling concepts.

UNIT IV

Real Time OS: OS services, Process management, Memory management device, File &IO subsystem management, Interrupt routines in RTOS, RTOS task scheduling models, Securities issues, RTOS mCOS-II & RTOS VxWorks.

UNIT V

Design Examples and Case Studies: Personal Digital Assistants, Digital thermometer, Case Studies of digital camera, Smart card, Case study of coding for sending application layer byte stream on TCP/IP network using RTOS VxWorks.

Text Books:

1. Frank Vahid/ Tony Givargis, ***"Embedded Systems Design"***, Wiley, 2002.
2. Raj Kamal, ***"Embedded Systems Architecture, and Programming"***, TMH Publication, 3rd Edition, 2015.
3. Andrew N. Sloss, ***"ARM System Developers Guide"***, ELSEVER Publication.

Reference Books:

1. Wayne Wolf, ***"Computer as Components – Principles of Embedded Computing System Design"***, Gulf Professional Publishing, 2nd Edition, 2008.
2. David E Simon, ***"An Embedded Software Primer"***, Addison Wesley Publication, 2004.

Course Title:	**IT Service Management**	**Semester V**	
Course Code	**BTITPE503B**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Infrastructure & Security Management**	**Credits**	**3**

Course Objectives:

1. To introduce practical implementation of Information Technology Service Management (ITSM).
2. To understand how an integrated ITSM framework can be utilized to achieve IT business integration, cost reductions and increased productivity.
3. To learn the best practices of ITSM methodology.

Course Outcomes:

After learning the course the students should be able:

1. To identify IT services as a means to provide functionality and value to customers.
2. To describe the needs and targets of the different stakeholders (service providers, customers, suppliers/partners) in the services value chain.
3. To demonstrate the value of a service management framework.
4. To explain the service management processes for given customers.
5. To select the appropriate tools to support a given designed service management solution.

Course Content:

UNIT I

IT Infrastructure: Introduction, Challenges in IT Infrastructure Management, Design Issues of IT Organizations and IT Infrastructure, IT System Management Process, IT Service Management Process, Information System Design Process.

UNIT II

Service Delivery Process: Service Level Management, Financial Management, IT Service Continuity Management, Capacity Management & Availability Management.

Service Support Process: Configuration Management, Incident Management, Problem Management, Change Management & Release Management.

UNIT III

Storage Management: Storage, Backup, Archive and Retrieve, Disaster Recovery, Space Management, Database and Application Protection and Data Retention.

UNIT IV

Security Management: Computer Security, Internet Security, Physical Security, Identity Management, Access Control System and Intrusion Detection.

UNIT V

Case Studies on how IT Service Management and ITIL processes make IT efficient and save cost for organizations.

Text Book:

1. Phalguni Gupta, Surya Prakash and Umarani Jayaraman, ***"IT Infrastructure & Its Management"***, Tata McGraw-Hill Education.

Reference Books:

1. W. Ronald Hudson, Ralph C. G. Haas, WaheedUddin, ***"Infrastructure Management: Integrating Design, Construction, Maintenance, Rehabilitation, and Renovation"***, McGraw-Hill, 1997. .

2. Anita Sengar, ***"IT Infrastructure Management"***, S.K. Kataria and Sons, 2nd Edition, 2009.

Course Title:	**Information Storage Management**	**Semester V**	
Course Code	**BTITPE503C**	**Course Type**	**Elective**
Pre-requisite	**Computer Architecture & Organization**	**L – T – P**	**3 – 0– 0**
Stream	**Information Management & Quality Control**	**Credits**	**3**

Course Objectives:

1. To evaluate storage architecture; understand logical and physical components of storage Infrastructure including storage subsystems.
2. To describe storage networking technologies such as FC-SAN, NAS, IP-SAN and data archival solution–CAS.
3. To identify different storage virtualization technologies and their benefits.
4. To understand and articulate business continuity solutions including, backup and recovery technologies, and local and remote replication solutions.
5. To define information security, and storage security domains and Identify parameters of managing and monitoring storage infrastructure and describe common storage management activities and solutions.

Course Outcomes:

After learning the course the students should be able:

1. To describe and apply storage technologies.
2. To identify leading storage technologies that provides cost-effective IT solutions for medium to large scale businesses and data centers.
3. To describe important storage technologies' features such as availability, replication, scalability and performance.
4. To design, analyze and manage clusters of resources.

Course Content:

UNIT I

Introduction to Information Storage Management - Intelligent Storage System (ISS) and its components Implementation of ISS as high-end and midrange storage-arrays, Direct Attached -Storage - Introduction to SCSI.

Introduction to parallel SCSI, SCSI Command Model – Storage Area Networks - Fiber Channel Connectivity, Login types, Topologies.

UNIT II

Storage networking technologies: Network-Attached Storage- General purpose servers vs. NAS Devices - Benefits of NAS, NAS File I/O – NAS Components, Implementation, File Sharing protocols, I/O operations – IPSAN-ISCSI, Components of ISCSI- Content-Addressed Storage.

UNIT III

STORAGE VIRTUALIZATION: Fixed Content and Archives, Types, Features, Benefits, CAS Architecture, object storage and Retrieval, examples - Storage Virtualization-forms of virtualization, SNIA Taxonomy – Storage virtualization configurations, challenges, Types of storage virtualization - Business Continuity- Overview of emerging technologies such as Cloud storage, Virtual provisioning, Unified Storage, FCOE, FAST.

UNIT IV

BUSINESS CONTINUITY AND RECOVERY: Information Availability, BC Terminology, Life cycle, Failure analysis - Backup and Recovery- Backup purpose, considerations, Backup Granularity, Recovery considerations- Backup methods, process, backup and restore operations , Overview of emerging technologies - duplication, offsite backup.

UNIT V

STORAGE SECURITY AND MANAGEMENT: Storage security framework, Securing the Storage infrastructure Risk triad - Managing the storage infrastructure, Monitoring the storage infrastructure, identify key parameters and components to monitor in a storage infrastructure List key management activities and examples Define storage management standards and initiative-Industry trend.

Text Book:

1. EMC Corporation, ***"Information Storage and Management"***, Wiley India, 1st Edition, 2009.

Reference Books:

1. IBM, ***"Introduction to Storage Area Networks and System Networking"***, 5th edition, November 2012.
2. Robert Spalding, ***"Storage Networks: The Complete Reference"***, Tata McGraw Hill, Osborne, 6th reprint 2003.
3. Marc Farley, ***"Building Storage Networks"***, Tata McGraw Hill, Osborne, 1st Edition, 2001.
4. Tom Clark, ***"Designing Storage Area Networks -A Practical Reference for Implementing Fiber Channel and IP SANs",*** Tata McGraw Hill 2003, 2nd edition.

Course Title:	**Network Management**	**Semester V**	
Course Code	**BTITPE503D**	**Course Type**	**Elective**
Pre-requisite	**Computer Networks & Internetworking Protocols**	**L – T – P**	**3 – 0– 0**
Stream	**Network**	**Credits**	**3**

Course Objectives:

1. To understand the principles of network management, different standards and protocols used in managing complex networks.
2. To understand the automation of network management operations and making use of readily available network management systems.

Course Outcomes:

After learning the course, the students should be able:

1. To acquire the knowledge about network management standards (OSI and TCP/IP).
2. To acquire the knowledge about various network management tools and the skill to use them in monitoring a network.
3. To analyze the challenges faced by Network managers.
4. To evaluate various commercial network management systems and open network Management systems.
5. To analyze and interpret the data provided by an NMS and take suitable actions.

Course Content:

UNIT I

Data communication and network management overview: Analogy of Telephone Network Management, Communications protocols and Standards, Case Histories of Networking and Management, Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions, Network and System Management, Network Management System Platform, Current Status and future of Network Management.

SNMPV1 Network Management Organization and Information Models, Managed network:. Managed network: Case Histories and Examples,

UNIT II

The History of SNMP Management, The SNMP Model, The Organization Model, System Overview, The Information Model.

SNMPV1 Network Management Communication and Functional Models: The SNMP Communication Model, Functional model. SNMP MANAGEMENT SNMPv2: Major Changes in SNMPv2, SNMPv2 System architecture, SNMPv2 Structure of Management Information, The SNMPv2 Management Information Base, SNMPv2 Protocol, Compatibility with SNMPv1.

SNMP MANAGEMENT RMON: What is Remote Monitoring? RMON SMI and MIB, RMON1, RMON2, ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON.

UNIT III

Telecommunication Management Network: Why TMN? Operations Systems, TMN Conceptual Model, TMN Standards, TMN Architecture, TMN Management Service Architecture, TMN Integrated View, Implementation.

UNIT IV

Network management tools and systems: Network Management Tools, Network Statistics Measurement Systems, Network Management systems, Commercial Network Management Systems.

UNIT V

Web-Based Management: NMS with Web Interface and Web-Based Management, Web Interface to SNMP Management, Embedded Web-Based Management, Desktop management Interface, Web-Based Enterprise Management, WBEM: Windows Management Instrumentation, Java management Extensions, Management of a Storage Area Network, Future Directions. Case Studies:

Text Book:

1. Mani Subrahmanian, ***"Network Management Principles and Practice"***, Pearson Education, 2nd Edition, 2010.

Reference Books:

1. Morris, ***"Network management"***, Pearson Education, 1st Edition, 2008.
2. Mark Burges, ***"Principles of Network System Administration"***, Wiley DreamTech, 1st Edition, 2008.

Course Title:	**Data Visualisation**	**Semester V**	
Course Code	**BTITPE503E**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Data Science**	**Credits**	**3**

Course Objectives:

1. Learn and understand the importance of data visualization.
2. Learn what is user experience in data visualization and its importance.
3. Learn about basic and advance chart types used in data visualization.
4. Learn the psychology of visualization with Gestalt Principles.

Course Outcome:

After learning the course, the student will be able to:

1. Get a solid understanding of how people work in data visualization project.

Course Content:

UNIT I

The seven stages of Data Visualization: Why data display requires planning, An example, Iteration and Combination, Principles. Getting Started with Processing: Sketching with processing, Example and Distributing your work, Examples and references, Functions,Sketching and Scripting
Mapping: Drawing a Map, Locations on map, Data on Map, Using your own data, Next step.

UNIT II

Time series: Milk, Tea, and Coffee (Acquire and parse), Cleaning the table (Filter and Mine), A simple plot (Represent and refine), Labeling the current data set (Refine and Interact), Drawing Axis labels(Refine), Choosing a proper representation (Represent and refine), Using rollovers to Highlights points(Interact), Ways to connect points(refine), Text labels as tabbed panes(Interact), Interpolation between data sets(Interact).

UNIT III

Connections and Correlations: Changing data sources, Problem statement, Preprocessing, Using the processed data (Acquire, Parse Filter and Mine), Displaying the results(Represent), Returning to the questions(Refine), Sophisticated sorting: Using salary as a Tiebreaker(Mine), Moving to multiple days(Interact), Smoothing out Interaction(Refine), Deployment Consideration (Acquire, Parse, filter).

UNIT IV

Scatterplot Maps: ++Preprocessing, Loading the data (Acquire and Parse), Drawing a scatterplot of Zip codes (Mine and represent), Highlighting Points while typing (Refine and Interact), Show the currently selected points(refine), Progressively Dimming and Brightening points(Refine), Zooming in (Interact), Changing How Points are Drawn when Zooming (Refine), Development issues (Acquire and Refine)

Trees, Hierarchies, and Recursion: Using recursion to build a Directory Tree, Using a Queue to Load Asynchronously (Interact), An improving the TreeMaps Display (Refine), Flying through files(Interact).

UNIT V

Networks and Graphs: A simple graph Demo, a more complicated Graph, Approaching Network Problem, Advanced graph example, Mining additional example.

Acquiring Data: Where to find data, Tools for Acquiring data from Internet, Loading files for use with processing, Loading text data, Dealing with files and folders, Listing files in folders, Asynchronous Image download, Using openStream() As a bridge to Java, Dealing with Byte arrays, Advanced web techniques, Using Databases, Dealing with large number of files. Parsing Data**:** Levels of efforts, Tools for gathering clues, Text is Best, Text Markup language, Regular expressions(regexps), Grammars and BNF Notations, Compressed Data, Vectors and Geometry, Binary data formats, Advanced detective work.

Text Book:

1. Ben Fry, ***"Visualizing Data: Exploring and Explaining data with Processing Environment"***, Shroff/O'Reilly Media, 2016

Reference Books:

1. Scott Murray, ***"Interactive Data Visualization for the web"***, Shroff/O'Reilly Media, 2016.
2. Julia Steele, Noah Lliinsky, ***"Designing Data Visualizations"***, Shroff/O'Reilly Media, 2012.
3. Kyran Dale, ***"Data Visualization with Python and JavaScript: Scrape, Clean, Explore &Transform your data"***, Shroff/O'Reilly Media, 2016.
4. Julia Steele, Noah Lliinsky, ***"Beautiful Visualization"***, Shroff/O'Reilly Media, 2016.

Course Title:	**Theory of Computation**	**Semester V**	
Course Code	**BTITOE504A**	**Course Type**	**Elective**
Pre-requisite	**Discrete Mathematics, Data Structure & Applications**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To understand problem classification and problem solving by machines.
2. To understand the basics of automata theory and its operations.
3. To study and compare different types of computational models.
4. Encourage students to study theory of computability and complexity.
5. To understand the P and NP class problems and its classification.
6. To understand the fundamentals of problem decidability and reducibility.

Course Outcomes:

After learning the course, students should be able:

1. To construct finite state machines to solve problems in computing.
2. To write mathematical expressions for the formal languages.
3. To apply well defined rules for syntax verification.
4. To construct and analyse Push Down Automata and Turing Machine for formal languages.
5. To express the understanding of the decidability and decidability problems.
6. To express the understanding of computational complexity.

Course Content:

UNIT I

Finite State Machines and Automata theory: Symbols, Strings, Language, Formal Language, Natural Language, Basic Machine and Finite State Machine. Definition and Construction-DFA, NFA, NFA with epsilon-Moves, Minimization Of FA, Equivalence of NFA and DFA, Conversion of NFA with epsilon moves to NFA, Conversion of NFA with epsilon moves to DFA, FSM with output

UNIT II

Regular Expressions: Definition and Identities of Regular Expressions, Construction of Regular Expression of the given L, Construction of Language from the RE, Construction of FA from the given RE using direct method, Conversion of FA to RE using Arden's Theorem, Pumping Lemma for RL, Closure properties of RLs, Applications of Regular Expressions.

UNIT III

Context free Grammar: Introduction, Formal Definition of Grammar, Notations, Derivation Process: Leftmost Derivation, Rightmost Derivation, derivation trees, Context Free Languages, Ambiguous CFG, Removal of ambiguity, Simplification of CFG, Normal Forms, Chomsky Hierarchy, Regular grammar, equivalence of RG (LRG and RLG) and FA.

UNIT IV

Push down Automata: Introduction and Definition of PDA, Construction (Pictorial/ Transition diagram) of PDA, Instantaneous Description and Acceptance of CFL by empty stack and final state, Deterministic PDA Vs Nondeterministic PDA, Closure properties of CFLs, pumping lemma.

Turing Machine: Formal definition of a Turing machine, Recursive Languages and Recursively Enumerable Languages, Design of Turing machines, Variants of Turing Machines: Multi-tape Turing machines, Universal Turing Machine, Nondeterministic Turing machines. Comparisons of all automata.

UNIT V

Undecidability and Computational Complexity: Decidability: Decidable problems concerning regular languages, Decidable problems concerning context-free languages, Un-decidability, Halting Problem of TM, A Turing-unrecognizable language. Reducibility: Un-decidable Problems from Language Theory, A Simple Un-decidable Problem PCP, Mapping Reducibility. Time Complexity: Measuring Complexity, The Class P, Examples of problems in P, The Class NP, Examples of problems in NP, NP-completeness.

Text Books:

1. Michael Sipser, "***Introduction to the Theory of Computation",*** CENGAGE Learning, 3rdEdition.
2. Vivek Kulkarni, ***"Theory of Computation",*** Oxford University Press.

Reference Books:

1. Hopcroft Ulman, "***Introduction to Automata Theory"***, Languages and Computations, Pearson Education Asia, 3rd Edition.
2. DanielI A. Cohen, ***"Introduction to Computer Theory"***, Wiley-India.
3. K.L.P Mishra, N. Chandrasekaran, ***"Theory of Computer Science (Automata, Languages and Computation)"***, Prentice Hall India, 3rd Edition.

Course Title:	**Graph Theory**	**Semester V**	
Course Code	**BTITOE504B**	**Course Type**	**Elective**
Pre-requisite	**Discrete Structures and Applications**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To understand and apply the fundamental concepts in graph theory.
2. To apply graph theory based tools in solving practical problems.
3. To improve the proof writing skills.

Course Outcomes:

After learning the course the students should be able to:

1. Solve problems using basic graph theory.
2. Identify induced sub graphs, cliques, matchings, covers in graphs.
3. Determine whether graphs are Hamiltonian and/or Eulerian.
4. Solve problems involving vertex and edge coloring.
5. Model real world problems using graph theory.

Course Content:

UNIT I

Basics- Graphs, Degree sequences, Distance in graphs, Complete, regular and bipartite graphs, Basic properties. Structure and Symmetry- Cut vertices, Bridges and blocks, Automorphism groups, Reconstruction problem.

UNIT II

Trees and connectivity - Properties of trees, Arboricity, Vertex and edge connectivity, Mengers theorem.

UNIT III

Eulerian and Hamiltonian graphs – Characterization of Eulerian graphs, Sufficient conditions for Hamiltonian graphs.

UNIT IV

Colouring and planar graphs - Vertex and edge colouring, Perfect graphs, Planar graphs, Euler's theorem, Kuratowski's theorem, Colouring of planar graphs, Crossing number and thickness.

UNIT V

External Graph theory - Turan's theorem, Ramsay's theorem, Szemeredi's regularity lemma, applications.

Text Books:

1. J. A. Bondy, U. S. R. Murthy, ***"Graph Theory"***, Springer Verlag, 2011.
2. D. B. West, ***"Introduction to Graph Theory"***, Pearson Education India, 2nd edition, 2011.

Reference Book:

1. R. Diestel, "***Graph Theory"***, Springer Verlag, 5th edition, 2017.

Course Title:	**Programming in Java**	**Semester V**	
Course Code	**BTITOE504C**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**3– 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
3. Be able to use the Java SDK environment to create, debug and run simple Java programs.

Course Outcomes:

After learning the course, the students should be able to:

1. Know the structure and model of the Java programming language.
2. Use the Java programming language for various programming technologies.
3. Develop software in the Java programming language (application).

UNIT I

Introduction to Java: Fundamentals of Object-oriented Programming, Evolution of Java, Overview of Java Language: Data types in Java, Operators and expressions, Decision Making and Branching: Control Statements such as If Else, Do statement, For statement, The Else if ladder, Jumps in loops, Labelled loops, While repetition statement, Switch statement, Break and continue statement, Arrays, Strings and Vectors: Creating one dimensional and multidimensional array, Strings, Vectors, Wrapper classes, Enumerated types, Annotations.

UNIT II

Object Oriented Programming: Classes , Objects And Methods: Defining class , Methods, Creating objects , Accessing Class members, Static Methods , Finalize Methods, Visibility Control, Method overloading, Method Overriding, Recursion. Interfaces, Constructors and finalizes Methods.
Packages and Applet Programming: Java API Packages, Using System Packages, Naming conventions, Creating Packages and Jar Files, Accessing and using a package, Hiding Classes, Applet Programming.

UNIT III

Multithreading: Creating threads, Extending Thread Class, Stopping and Blocking a thread, Life cycle of a thread, Using thread method, Thread exceptions, Implementing the Run able interface, Interthread communication. Managing Errors and Exceptions: Types of errors, Exceptions, Syntax of exception handling code, Multiple catch statements, Throwing your own exception, Using exceptions for debugging.

UNIT IV

Graphics Programming: The Graphics class, Lines and Rectangles, Circles, Arc and ellipses, Polygons, Drawing Bar charts, AWT Package and Swings.

UNIT V

Managing Files & I/O Handling: Files and Streams, Stream classes, Byte Stream Classes , Character Stream Classes, Using Streams, Reading / writing bytes and characters , Interactive Input and Output, Other Stream classes.

Text Books:

1. E. Balagurusamy, **"*Programming with Java – A Primer*"**, Tata – McGraw-Hill Publication, 4th Edition, 2010.
2. Steven Holzner et al. **"*Java 2 Programming*"**, Black Book, Dreamtech Press, 2009.

Reference Books:

1. H.M. Deitel, P.J. Deitel, **"*Java - How to Program*"**, PHI Publication, 6th Edition, 2005.
2. Bruce Eckel, **"*Thinking in Java*"**, PHI Publication.
3. Game Theory Tim Lindholm, Frank Yellin, Bill Joy, Kathi Walrath, **"*The Java Virtual Machine Specification*"**, Addison Wesley Publication.

Course Title:	**Human Computer Interaction**	**Semester V**	
Course Code	**BTITOE504D**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

Upon successful completion of this course, students should be able to:

1. Design, implement and evaluate effective and usable graphical computer interfaces.
2. Describe and apply core theories, models and methodologies from the field of HCI.
3. Describe and discuss current research in the field of HCI.
4. Implement simple graphical user interfaces using the Java Swing toolkit.
5. Describe special considerations in designing user interfaces for older adults.

Course Outcomes:

After learning the course, the students gain :

Knowledge

1. Describe and apply core theories, models and methodologies from the field of HCI.
2. Describe what the user-centered design cycle is and explain how to practice this approach to design interactive software systems.
3. Analyze one after another the main features of interactive systems, and explain how to gauge the usability of digital environments, tools and interfaces.

Professional Skill

1. Conduct user and task analysis.
2. Implement graphical user interfaces with modern software tools.
3. Critique and evaluate interactive software using guidelines from human factor theories.

Course Content:

UNIT I

Introduction: The human, The computer, The interaction, Paradigms, Usability of Interactive Systems, Guidelines, Principles, and Theories.
Design Process- Interaction design basics, HCI in the software process, Design rules, Implementation support, Evaluation techniques, Universal design, User support.

UNIT II

Models and Theories, Cognitive models, Socio-organizational issues and stakeholder requirements, Communication and collaboration models, Task analysis, Dialogue notations and design, Models of the system, Modeling rich interaction.

UNIT III

Interaction Styles- Direct Manipulation and Virtual Environments, Menu Selection, Form Filling and Dialog Boxes, Command and Natural Languages, Interaction Devices, Collaboration and Social Media Participation.

UNIT IV

Design Issues- Quality of Service, Balancing Function and Fashion, User Documentation and Online Help, Information Search, Information Visualization.

UNIT V

Outside the Box- Group ware, Ubiquitous computing and augmented realities, Hypertext, multimedia, and the World Wide Web.

Text Books:

1. Alan Dix, Janet Finlay, ***"Human Computer Interaction"***, Pearson Education, 3rd edition, 2009.
2. Ben Shneiderman, ***"Designing the User Interface - Strategies for Effective Human Computer Interaction"***, Pearson Education, 2010.

Reference Books:

1. M. B. Rosson, J. M. Carroll***"Usability Engineering: Scenario-Based Development of Human-Computer Interaction"***, Elsevier, 2002.
2. Alan Cooper, ***"The Essentials of Interaction Design"***, Wiley Publishing, 2007.
3. Nielsen, J. Morgan Kaufmann, San Francisco, **"*Usability Engineering"***, 1993.
4. Heim, S., **"*The Resonant Interface: HCI Foundations for Interaction Design"***, Addison-Wesley, 2007.

Course Title:	**Game Theory**	**Semester V**	
Course Code	**BTITOE504E**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives

1. The course is intended for students and teachers of institutions which offer undergraduate engineering programmes.
2. The aim of the course is to provide an introduction to the study of game theory which has found wide applications in economics, political science, sociology, engineering apart from disciplines like mathematics and biology
3. The course would introduce to the fundamental tools of game theory, a few equilibrium concepts, apart from numerous exercises and applications
4. Knowledge of game theory would help students to understand and analyse real life situations such as market behaviour or voting in elections, apart from equipping them with analytical concepts which might be useful should they decide to pursue social sciences, engineering, sciences or managerial higher studies
5. This is an interdisciplinary course, hence not only social sciences but science and engineering departments of different universities can benefit from it
6. The six modules of the course are as follows,
   a. Introduction to Game Theory
   b. Strategic Games and Nash Equilibrium
   c. Illustrations of Nash Equilibrium
   d. Mixed Strategy Nash Equilibrium
   e. Extensive Games and Nash Equilibrium
   f. Illustrations of Extensive Games and Nash Equilibrium

Course Outcomes:

After learning the course the students should be able to:

1. Solve problems using basic graph theory.
2. Identify induced subgraphs, cliques, matchings, covers in graphs.
3. Determine whether graphs are Hamiltonian and/or Eulerian.
4. Solve problems involving vertex and edge coloring.
5. Model real world problems using graph theory.

Course Content:

UNIT I

Introduction to Game Theory: Concept of game theory, Theory of rational choice, Interacting decision makers. Strategic Games and Nash Equilibrium: Strategic games: examples, Nash equilibrium: concept and examples, Best response functions, Dominated Actions, Symmetric games and symmetric equilibrium.

UNIT II

Illustrations of Nash Equilibrium: Cournot's model of duopoly market, Bertrand's model of duopoly market, Electoral Competition, War of Attrition, Auctions, Accident Laws.

UNIT III

Mixed Strategy Nash EquilibriumL: Introduction, Strategic games with randomisation, Mixed strategy Nash equilibrium: concept and examples, Dominated Actions, Formation of Players' beliefs.

UNIT IV

Extensive Games and Nash Equilibrium: Introduction to extensive games, Strategies and outcomes, Nash equilibrium, Subgame perfect Nash equilibrium, Backward induction

UNIT V

Illustrations of Extensive Games and Nash Equilibrium: Stackelberg model of duopoly markets, Ultimatum game

Text books:

1. Osborne, M.J. , "***An Introduction to Game Theory***", Oxford University Press, 2004
2. Mas-Colell, A., M.D. Whinston and J.R. Green, "***Microeconomic Theory***", Oxford University Press, 1995
3. Gibbons, R., "***A Primer in Game Theory***", Pearson Education, 1992

Course Title:	**Constitution of India**	**Semester V**	
Course Code	**BTHM501**	**Course Type**	**Audit**
Pre-requisite	**Nil**	**L – T – P**	**2 – 0 – 0**
Stream	**Nil**	**Credits**	**-**

Course Objectives:

1. To enable the student to understand the importance of constitution.
2. To understand the structure of ececutive, legislature and judiciary.
3. To understand philosophy of fundamental rights and duties.
4. To understand autonomous nature of constitutional bodies like Supreme court and high court, Controller and aditional general of India and Election commission of India.
5. To understand central and state relation , financial and administrative.

Course Outcomes:

After learning the course, the student will be able to:

1. Understand historical background of constitution making and its importance for building a democratic India, the structure of Indian government, the structure of state government, the local administration.
2. Apply the knowledge on directive principle of state policy, the knowledge in strengthsening of the constitutional institutions like CAG, Election Commission and UPSC for sustaining democracy.
3. Analyze the history, features of Indian Constitution, the role of Governer and Chief misister, role of state election commision, the decentralization of power between central, state and local self-government.
4. Able to evaluate Preamle, Fundamental Rights and Duties, Zilla Panchayat, block level organization, various commisions viz SC/ST/OBC and women.

Course Content:

UNIT I

Introduction to Indian Constitution Historical background, Philosophy of Indian Constitution, Preamble of Constitution- its forms and Importance, Features of Indian Constitution, The nature of Indian Federation.

UNIT II

Fundamental Rights and Directive Principles Fundamental Rights- its forms and importance, Fundamental rights in Constitution, Evaluation of Fundamental rights, Fundamental duties, Directive Principles of State Policies(Meaning, Objectives and Source), Classification of Directive Principles, Implementation of Directive Principles.

UNIT III

Composition and Structure of Parliament Function of Parliament, Law making Procedure, Executive Council structure and Role, State assembly, Changing Trends of Parliament.
Judiciary and Election Commission Forms of Judiciary, Power, Function and Role of Supreme Court, Judicial Review, Judicial Activism, Structure, Function and Role of Election Commission, Electoral System and Reforms in it.

UNIT IV

Socialism of Constitution Provision for Women Empowerment, Protection of Rights of Backward Class, Special Provision for Scheduled Tribes, Protection of Rights of workers, Socialistic democracy.

UNIT V

Democracy in India: Challenges Constitutional Institutions and their role, Lokpal and Lokayukt, State Central Relation, Important Amendments, Nationalism, Criminalisation of Politics**.**

Text Books:

1. D. D. Basu, "Introduction to the Constitution of India", LexisNexis Publishers, 23rd Edition, 2018.
2. B. Shiva Rao (Editor), "Framing of Indian's Constitution, Select Documents", Vol. 1, 2015.

Reference Books:

1. T. K. Tope, "Constitutional Law of India", Sujata V. Manohar (Editor), Eastern Book Company, 3rd Edition, 2010.
2. Sir Ivor Jennings, "Some Characteristics of Indian Constitution", Geoffrey Cumberlege Publishers, 1953.

Course Title:	**Computer Networks and Internetworking Protocols Lab**	**Semester V**	
Course Code	**BTITL505**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Conversion of IP addresses
   (e.g. I/P: 10.24.164.254 O/P: 00001010.00011000.10000000.11111110 and I/P:binary dotted O/P: decimal dotted)
2. Introduction to Wireshark
3. Wireshark Lab: Ethernet and ARP
4. Wireshark Lab: IP
5. Wireshark Lab: ICMP, study of ping and traceroute command
6. Wireshark Lab: UDP
7. Wireshark Lab: TCP
8. Study of ftp, telnet tools and network configuration files
9. DHCP server configuration
10. Socket programming for UDP and TCP.

Course Title:	**Software Engineering Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credit**	**1**

List of Experiments:

Part: I RDBMS

1. To develop a mini project for an RDBMS, the following exercise have been specified to give idea/prerequisite learning for the concept required in defining the problem statement for an RDBMS.
2. Design and draw an ER/EER diagram and map this diagram to the database tables.
3. Create database tables for the problem. Perform add, insert, delete, update operations.
4. Use DDL statements and apply all constraints on tables to make the operations on tables.
5. Write and execute triggers and procedures/functions.
6. Generate a simple report.

Part: II

Following exercise has been specified to give idea/prerequisite learning for the concept in defining the problem statement for a front end to RDBMS based system.

Front end tools support developments of the following concept:

1. Controls
2. Properties for every control of the form
3. Events
4. Programming components
5. Proper interface to the back end database

The desktop database can be chosen from MSSQL, ORACLE, mySQL or equivalent databases packages. The front end development tools can be chosen from VB6, .NET, JAVA or equivalent tools.

Part: III

The statement of the problem will be the mini project for the group. The design of the project shall follow the software development life cycle. It should prepare a report for each stage (this will be the part of project manual later).

The group should understand and prepare proper documentation in relation with following,

1. Problem definition in detail.
2. Literature survey.
3. Requirement analysis.

4. System analysis (Draw Level 2 DFD at least).
5. System design
6. Implementation
7. Use cases
8. Testing

While designing the project the care should be taken to follow the coding conventions, software project design standards, data dictionary, etc. Staff in-charge will frame the mini project specification to be performed by group of students. There will be different problem definition to each group. The students will prepare an installable CD for the mini projects; *README* file will have the project description, system requirements, development details, and installation instruction. *User manual* will have the interaction screens and the way to use the developed project.

Course Title:	**Embedded Systems Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Software Application & Development**	**Credit**	**1**

Lab Experiments List:

1. Study of ARM evaluation system
2. Interfacing ADC and DAC.
3. Interfacing LED and PWM.
4. Interfacing real time clock and serial port.
5. Interfacing keyboard and LCD.
6. Interfacing EPROM and interrupt.
7. Mailbox.
8. Interrupt performance characteristics of ARM and FPGA.
9. Flashing of LEDS. .
10. Implementing zigbee protocol with ARM.

Course Title:	**IT Service Management Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Infrastructure & Security Management**	**Credit**	**1**

Lab Experiments List:

1. To study the Information System Design Process.
2. To study the relationship of service level management with other service delivery processes.
3. To study the Problem, Change and Incident Management.
4. To study and demonstrate disaster recovery.
5. To study and demonstrate the various security techniques used to secure the data while transmitting over the internet.

Course Title:	**Information Storage Management Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Compulsory**
Prerequisite	**Computer Architecture & Organization**	**L – T – P**	**0– 0 – 2**
Stream	**Information Management & Quality Control**	**Credit**	**1**

Lab Experiments List:

1. Data Center Environment Data Center Environment
   a) Install the VNXe Simulator
   b) Discover the infrastructure
2. Intelligent storage system
   a) Navigate the storage system
   b) Create a block device
   c) Create a file device
3. FC SAN
   a) FC san configuration
   b) FC san trace
4. IP SAN
   a) IP SAN configuration
   b) ISCSI san trace
5. Host-based business continuity
   a) Multipath
6. Managing protection services
   a) Array-based protection
   b) Configuring LUN protection
7. Managing storage infrastructure
   a) Monitoring and reporting

Course Title:	**Network Management Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Elective**
Prerequisite	**Computer Architecture & Organization**	**L – T – P**	**0– 0 – 2**
Stream	**Network**	**Credit**	**1**

Lab Experiments List:

1. Network Monitoring tools
   a) Status b)Route c)Traffic Tools

2. Monitoring and management network using SNMP
   a) Basic SNMP, b) Advanced SNMP v3 Authentication/Encryption and ACL , c) SNMP Trap Daemon Implementation

3. Install and configure SNMP MIB browser
   a) qtmib b)snmpB c) OpManager MIB browser

4. Network Statistics and measurement
   a) LAN Traffic Monitoring b) Protocol statistics

5. LAN Troubleshooting using Wireshark

6. To study log system using open source tools.

7. Study of commercial network management tools: HPOpenView, OpManager, GFILanguard and IBM NMS.

Course Title:	**Data Visualization Lab**	**Semester V**	
Course Code	**BTITL506**	**Course Type**	**Compulsory**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Data Science**	**Credit**	**1**

Lab Experiments List:

1. To learn about area plots and how to create them with Matplotlib, histograms and how to create them with Matplotlib,
2. To learn about bar charts, and how to create them with Matplotlib, pie charts, and how to create them with Matplotlib
3. To learn box plots and how to create them with Matplotlib, and scatter plots and bubble plots and how to create them with Matplotlib.
4. To learn about advanced visualization tools such as waffle charts and word clouds and how to create them.
5. To learn about seaborn, visualization library, and how to use it to generate attractive regression plots.
6. To learn about Folium, visualization library, designed especially for visualizing geospatial data.
7. To learn how to use Folium to create maps of different regions of the world and how to superimpose markers on top of a map, and how to create choropleth maps.

Course Title:	**Operating Systems**	**Semester VI**	
Course Code	**BTITC601**	**Course Type**	**Compulsory**
Pre-requisite	**Nil**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To study the basic concepts and functions of operating systems.
2. To understand the structure and functions of OS.
3. To learn about Processes, Threads and Scheduling algorithms.
4. To understand the principles of concurrency and Deadlocks.
5. To learn various memory management schemes.
6. To study I/O management and File systems.

Course Outcomes:

After learning the course, the students should be able:

1. To design various Scheduling algorithms.
2. To apply the principles of concurrency.
3. To design deadlock, prevention and avoidance algorithms.
4. To compare and contrast various memory management schemes.
5. To design and Implement a prototype file system.

Course Content:

UNIT I

Operating System Structures: Definition, Types of operating system, Real time operating system, System components, Sys-tem services, Systems calls, System programs, System structure, Virtual machines, System design and implementation.

UNIT II

Processes and CPU scheduling: Process concept, Process scheduling, Operation on a process, Co-operating processes, Threads, Interprocess communication, Scheduling criteria, Scheduling algorithms, Multiple-processor scheduling, Real-time scheduling, Scheduling algorithms and performance evaluation.

Process Synchronization: The critical-section problem, Critical regions, Synchronization hardware, Semaphores, Classical problems of synchronization, Monitors.

UNIT III

Deadlocks: Systems model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock, Combined approach to deadlock handling.

UNIT IV

Memory Management and Virtual Memory: Logical versus physical address space, Swapping, Contiguous allocation, Paging, Segmentation with paging, Demand paging, Page replacement algorithms, Thrashing.

UNIT V

File Management: File system and secondary storage devices, Real-time operating systems.

Text Books

1. A. Silberschatz, P. Galvin, ***"Operating System Concepts"***, Wiley Publication, 9th Edition, 2013.
2. A. S. Tanenbaum, H. Bos, ***"Modern Operating Systems"***, Pearson Education, 4th Edition, 2016.

Reference Books:

1. D.M. Dhamdhere, ***"Systems Programming and Operating Systems"***, Tata McGraw Hill Publication, 2nd Edition, 2001.
2. G. Nutt, ***"Operating Systems Concepts"***, Addison Wesley Publication, 3rd Edition.
3. H. M. Deitel, ***"An Introduction to Operating Systems"***, Pearson education Publication, 3rd edition, 2007.

Course Title:	**Database Management Systems**	**Semester V**I	
Course Code	**BTITC602**	**Course Type**	**Compulsory**
Pre-requisite	**Nil**	**L – T – P**	**3 – 1– 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To understand architecture and functioning of database management systems.
2. To learn relational mode.
3. To use structured query language (SQL) and its syntax, transactions, database recovery and techniques for query optimization.
4. To acquaint with various normalization forms and query processing.
5. To learn indexing methods.

Course Outcomes:

1. After learning the course the students should be able:
2. To explain need of database management.
3. To design and implement a database schema for a given problem-domain.
4. To normalize a database.
5. To create and query a database using SQL DML/DDL commands, stored procedures and functions.
6. To declare and enforce integrity constraints on a database.
7. To illustrate understanding of indexing methods.

Course Content:

UNIT I

Introduction: Basic concepts, Advantages of DBMS over file-processing systems, Data abstraction, Data models and data independence, Components of DBMS and overall structure of DBMS, Data modeling, Entity, Attributes, Relationships, Constraints, Keys E-R diagrams, Components of E-R Model.

UNIT II

Relational Model: Basic concepts, Attributes and domains, Concept of integrity and referential constraints, Schema diagram. Relational query languages, Relational Algebra and Relational Calculus: Tuple relational and domain relational calculus.

Structured Query Language-I: Introduction, Characteristics and advantages, Data types and literals, DDL, Tables: creating, modifying, deleting.

UNIT III

Views: creating, dropping, Updation using views, DML, Operators, SQL DML queries, SELECT query and clauses.

Structured Query Language- II: Set operations, Predicates and joins, Set membership, Tuple variables, Set comparison, Ordering of tuples, Aggregate functions, Nested queries, Database modification using SQL Insert, Update and Delete queries, Dynamic and SQL and concept of stored procedures, Query-by-example.

UNIT IV

Relational Database Design: Notion of normalized relations, Functional dependency, Decomposition and properties of decomposition, Normalization using functional dependency, Multi-valued dependency and join dependency. Storage and File Systems: Secondary storage, RAID, File organization, Indices, Static and dynamic hashing, B-Trees and B+ Trees.

UNIT V

Query Processing and Transaction Management: Measures of query cost, Selection operation, Sorting and join operation, Transaction concept, Components of transaction management, Concurrency and recovery sys-tem, Different concurrency control protocols such as timestamps and locking, Validation, Multiple granularity, Deadlock handling, Different crash recovery methods such as log-based recovery, Shadow-paging, Buffer management and Remote backup system.

Text Books:

1. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, ***"Database System Concepts"***, McGraw Hill Education, 6th Edition, 2011.
2. RamezElmasri and Shamkant B. Navathe, ***"Fundamental Database Systems"***, Pearson Education, 7th Edition, 2015.
3. Raghu Ramkrishnan, Johannes Gehrke, ***"Database Management Systems"***, McGraw Hill Education, 3rd Edition, 2007.

Reference Books:

1. Carlos Coronel, Steven Morris ***"Database systems: Design Implementation andManagement"***, Cengage Learning Press, 11th Edition, 2014.
2. J. Murach, ***"Murach's MySQL"***, Shroff Publication, 2nd Edition, 2016.
3. J. Murach, ***"Murach's Oracle SQL and PL/SQL: Works with All Versions Through 11g"***, Shroff Publication, 2008.

Course Title:	**Software Testing**	**Semester VI**	
Course Code	**BTITPE603A**	**Course Type**	**Elective**
Pre-requisite	**Software Engineering**	**L – T – P**	**3 – 0 – 0**
Stream	**Software Application & Development**	**Credits**	**3**

Course Objectives:

1. To study fundamental concepts in software testing, including software testing objectives, processes, criteria, strategies, and methods.
2. To learn planning of a test project, designing test cases and test data, conducting test operations, managing software problems and defects, and generating a test report.
3. To develop an understanding of the meaning and importance of quality in relation to software systems and the software development process.
4. To study issues and techniques for implementing and managing software quality assurance processes and procedures.

Course Outcomes:

After learning the course, the students should be able:

1. To apply software testing knowledge and its processes to software applications.
2. To identify various software testing problems.
3. To solve software testing problems by designing and selecting software test models, criteria, strategies and methods.
4. To apply the techniques learned to improve the quality of software development.
5. To prepare a software quality plan for a software project.

Course Content:

UNIT I

Principles of Testing Software development life cycle model: Phases of software project, Quality, Quality assurance and quality control, Testing, Verification and validation, Process models to represent various phases, Life cycle models, Software testing life cycle.

White Box Testing (WBT) and Black Box Testing: Static testing, Structural testing, Challenges in WBT. Black box testing: Black box testing process.

UNIT II

Integration Testing: Definition, As a type of testing: Top-down integration, Bottom-up integration, Bi-directional integration, System integration, Choosing integration method, As a phase of testing, Scenario testing: System scenarios, Use case scenarios, Defect bash.

UNIT III

System and Acceptance Testing, Functional Vs non Functional, Functional system testing, Non-functional system testing, Acceptance testing.

UNIT IV

Performance testing, Regression testing, Internationalization testing, Adhoc testing. Factors governing performance of testing, Methodology, tools and process for performance testing. Regression Testing: Introduction, Types of Regression testing, Regression testing process. Adhoc testing: Introduction, Buddy testing, Pair testing, exploratory testing, Iterative testing, Agile and Extreme testing, XP work flow, Defect seeding.

UNIT V

Testing Object Oriented Software: Introduction, Comparison of object oriented and procedural software, Sys-tem testing example, Unit testing of classes, Tools for testing object oriented software, Testing web applications.

Text Book:

1. SrinivasanDesikan, Gopalaswamy Ramesh, ***"Software Testing: Principles and Practices"***, Pearson publication, 2nd Edition, 2006.

Reference Books:

1. LoiseTamres, ***"Introducing Software Testing"***, Pearson publication, 2002.
2. Boris Beizer, ***"Software Testing Techniques"***, Dreamtech press, 2nd Edition, 2014

Course Title:	**Data Storage Technologies & Networks**	**Semester VI**	
Course Code	**BTITPE603B**	**Course Type**	**Elective**
Pre-requisite	**Computer Network & Internetworking Protocols, Operating Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Infrastructure & Security Management**	**Credits**	**3**

Course Objectives:

1. To gain knowledge and understand the design of a Data Centre.
2. To understand the best practice of design in the Data Centre.
3. To learn the options in the running of an efficient Data Centre.
4. To understand the value of data to a business, Information Lifecycle.
5. To understand the challenges in data storage and data management.
6. To learn solutions available for data storage.

Course Outcomes:

After learning the course, the students should be able:

1. To explain the design of a data center and storage requirements.
2. To discuss the various types of storage and their properties.
3. To explain physical and virtualization of storage.
4. To explain the backup, archiving with regard to recovery and business continuity.

Course Content:

UNIT I

Data Centre: Introduction, Site Selection and Environmental Considerations, Hierarchical or Layered Architecture, Architect Roles, Goals and Skills, Architecture Precursors.

Data Centre Design: Architecture Design and Standards Recommendations, Raised Access Floor and Design Best Practices, connecting the infrastructure with copper and fiber. IT Hardware, Cooling System Options and Environmental Control, Electrical Power Systems, Room Layout, Fire Protection and Security Systems, Building Automation and Energy Management Systems, Commissioning and Handover.

UNIT II

Storage Management: Introduction to Storage Technology, Storage Systems Architecture, Physical and logical components of a connectivity environment, Major physical components of a disk drive and their functions, Concept of RAID and its components, Different RAID levels and their suitability for different application environments: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5, RAID 0+1, RAID 1+0, RAID 6, Integrated and Modular storage systems, high-level architecture and working of an intelligent storage systems.

UNIT III

Networked Storage: Evolution of networked storage, Architecture, components, and topologies of FC-SAN, NAS, and IP-SAN, Benefits of the different networked storage options, Need for long-term archiving solutions and describe how CAS fulfill the need, Appropriateness of the different networked storage options for different application environments

UNIT IV

Managing Data Center: Reasons for planned/unplanned outages, Impact of downtime, Difference between business continuity (BC) and disaster recovery (DR), RTO and RPO, Identification of single points of failure in a storage infrastructure and solutions to mitigate these failures, Architecture of backup/recovery and the different backup/recovery topologies, replication technologies and their role in ensuring information availability and business continuity Remote replication technologies and their role in providing disaster recovery and business continuity capabilities, Key areas to monitor in a data center, Industry standards for data center monitoring and Management Key metrics to monitor storage infrastructure.

UNIT V

Securing Storage and Storage Virtualization: Information Security, Critical security attributes for information systems, Storage security domains, Analyze the common threats in, each domain, Storage Virtualization: Forms, Configurations and Challenges, Types of Storage Virtualization: Block-level and File-Level.

Text Books:

1. Mauricio Arregoces, ***"Data Center Fundamentals"***, Cisco Press, 1st edition, 2003.
2. Robert Spalding, ***"Storage Networks: The Complete Reference"***, Tata McGraw Hill, Osborne, 2003.
3. Marc Farley, ***"Building Storage Networks"***, Tata McGraw Hill, Osborne. 2001.
4. Meeta Gupta, ***"Storage Area Network Fundamentals"***, Pearson Education Limited, 2002

Reference Books:

1. G. Somasundaram, AlokShrivastava, ***"Information Storage and Management"***, EMC Education Series, Wiley Publishing Inc., 2011.
2. Gustavo Santana, ***"Data Center Virtualization Fundamentals: Understanding Techniques and Designs for Highly Efficient Data Centers with Cisco Nexus, UCS, MDS, and Beyond"***, Cisco Press, 1st Edition, 2013

Course Title:	**Service Oriented Architecture**	**Semester VI**	
Course Code	**BTITPE603C**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Information Management & Quality Control**	**Credits**	**3**

Course Objectives:

1. To gain understanding of the basic principles of service orientation.
2. To learn service oriented analysis techniques.
3. To learn technology underlying the service design.
4. To learn advanced concepts such as service composition, orchestration and Choreography.
5. To know about various WS specification standards.

Course Outcomes:

After learning the course, the students should be able:

1. Build applications based on XML.
2. Develop web services using technology elements.
3. Build SOA-based applications for intra-enterprise and inter-enterprise applications.

Course Content:

UNIT I

Introducing SOA: Fundamental SOA: Common Misperceptions about SOA, Common tangible benefits of SOA, Common pitfalls of adopting SOA, The Evolution of SOA:-from XML to Web services to SOA, The continuing evolution of SOA, The roots of SOA. Web Services and Primitive SOA: The Web services framework-Services, Service descriptions, messaging with SOAP.

UNIT II

Web Services and Contemporary SOA: Message exchange patterns- Service activity-coordination-Atomic transactions-Business activities-Orchestration-Choreography- Web Services and Contemporary SOA: Addressing- Reliable messaging-Correlation- Policies- Metadata exchange- Security-Notification and eventing.SOA and Service-Orientation: Principles of Service - Anatomy of a service-oriented architecture- Common principle of service orientation-Service Layers –Service orientation.

UNIT III

Building SOA: SOA Delivery Strategies- SOA delivery lifecycle phases. Service-Oriented Analysis: Introduction to service-oriented Analysis-Benefits of a business-centric SOA- Deriving business services-Service-Oriented Analysis: Service modeling, Service modeling guidelines- Classifying service model logic- Contrasting service modeling approaches.

UNIT IV

Service-Oriented Design: Introduction to service-oriented design- WSDL-related XML Schema language basics- WSDL language basics- SOAP language basics- Service interface, design tools. SOA Composition Guidelines: Steps to composing SO Considerations for choosing service layers and SOA standards, positioning of cores and SOA extensions.

UNIT V

SOA Service Design: - Overview-Service design of business service, application service, task centric service and guidelines. SOA Business Process Design: WS-BPEL language basics-WS Coordination.

SOA support in J2EE – Java API for XML-based web services (JAX-WS) - Java architecture for XML binding (JAXB) – Java API for XML Registries (JAXR) - Java API for XML based RPC (JAX-RPC)-Web Services Interoperability Technologies (WSIT)

Text Books

1. Thomas Erl, ***"Service-Oriented Architecture: Concepts, Technology, and Design"***, Pearson Education, 2006.
2. Frank. P. Coyle, ***"XML, Web Services and The Data Revolution"***, Pearson Education, 2002.
3. SandeepChatterjee, James Webber, ***"Developing Enterprise Web Services. An Architect's Guide"***, Pearson Education, 2005.
4. Eric Newcomer, Greg Lomow, ***"Understanding SOA with Web Services"***, Pearson Education, 2005.
5. Ron Schmelzer et al. ***"XML and Web Services"***, Pearson Education, 2002

Reference Books:

1. Dan woods and Thomas Mattern, ***"Enterprise SOA designing IT for Business Innovation"***, O'REILLY, 1st Edition, 2006.
2. James McGovern, Sameer Tyagi, Michael E. Stevens, Sunil Mathew, ***"Java Web. Services Architecture"***, Morgan Kaufmann Publishers, 2003.
3. AtulKahate, ***"XML and Related technologies"***, Pearson Education, 2008.
4. Kennard Scibner and Mark C. Stiver, ***"Understanding SOAP"***, SAMS publishing.
5. B. V. Kumar, S. V. Subrahmanya, ***"Web Services: An Introduction"***, TMH India, 2nd Edition, 2012.

Course Title:	**Network Programming**	**Semester VI**	
Course Code	**BTITPE603D**	**Course Type**	**Elective**
Pre-requisite	**Computer Network & Internetworking Protocols, Operating Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Network**	**Credits**	**3**

Course Objectives:

1. To learn the basics of socket programming using TCP Sockets.
2. To learn about Socket options.
3. To learn to develop Macros for including Objects in MIB Structure.
4. To understand SNMPv1, v2 and v3 protocols & practical issues.

Course Outcomes:

After learning the course, the students should be able:

1. To analyze the requirements of a networked programming environment and identify the issues to be solved.
2. To create conceptual solutions to those issues and implement a programming solution.
3. To understand the key protocols those support the Internet.
4. To apply several common programming interfaces to network communication.
5. To understand the use of TCP/UDP Sockets.
6. To apply advanced programming techniques such as Broadcasting, Multicasting.

Course Content:

UNIT I

Socket and Application Devlopment: Introduction to Socket Programming, System Calls, Address conversion functions, POSIX Signal Handling, Server with multiple clients, Boundary conditions, Server process Crashes, Server host Crashes, Server Crashes and reboots, Server Shutdown, I/O Multiplexing, I/O Models, TCP echo client/server with I/O Multiplexing

UNIT II

Socket Option: Socket options, getsockopt and setsockopt functions, Generic socket options, IP socket options, ICMP socket options, TCP socket options, Multiplexing TCP and UDP sockets, SCTP Sockets, SCTP Client/server, Streaming Example, Domain name system, gethostbyname, gethostbyaddr, getservbyname and getservbyport functions, Protocol Independent functions in TCP Client/Server Scenario.

UNIT III

Advanced Socket: IPv4 and IPv6 interoperability, Threaded servers, Thread creation and termination, TCP echo server using threads, Mutex Condition variables, Raw sockets, Raw socket creation, Raw socket output, Raw socket input, ping program, traceroute program

UNIT IV

Simple Network Management: SNMP network management concepts, SNMPv1 Management information, MIB Structure, Object syntax, Standard MIB's, MIB-II Groups, SNMPv1 protocol and Practical issues.

SNMP V2, V3 and RMO: Introduction to SNMPv2, SMI for SNMPV2 Protocol, SNMPv3 Architecture and applications, Security and access control model, Overview of RMON.

UNIT V

Protocols, Sessions, State, and Implementing Custom Protocols State vs. Stateless, Methods for Maintaining State, What Is a Protocol? Designing a Custom Protocol, Our Chat Protocol, Protocol Registration, Elementary Name, Address Conversions and design decisions Domain Name System, gethostbyname Function, RES_USE_INET6 Resolver Option, gethostbyname2 Function and IPv6 Support, gethostbyaddr Function, uname Function, gethostname Function, getservbyname and getservbyport Functions

Text Books:

1. W. Richard Stevens, ***"UNIX Network Programming Vol-I"***, Addison-Wesley Professional, 3rd Edition,2003.
2. William Stallings, ***"SNMP, SNMPv2, SNMPv3 and RMON 1 and 2"***, Pearson Edition, 3rd Edition, 2009.

Reference Book:

1. D.E. Comer, ***"Internetworking with TCP/IP Vol- III: Client-Server Programming and Application BSD Sockets Version"***, Pearson Edition, 2nd Edition, 2003.

Course Title:	**Data Warehousing and Data Mining**	**Semester VI**	
Course Code	**BTITOE603E**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Data Science**	**Credits**	**3**

Course Objectives:

1. Introduce the concepts, techniques, design and applications of data warehousing and data mining.
2. Enable students to understand and implement classical algorithms in data mining and data warehousing.
3. Enable students to learn how to analyze the data, identify the problems and choose the relevant algorithms to apply.

Course Outcomes:

After learning the course the student will be able:

1. Understand the functionality of the various data mining and data warehousing components.
2. Appreciate the strengths and limitations of various data mining and data warehousing models.
3. Compare the various approaches to data warehousing and data mining implementations.
4. Describe and utilize a range of techniques for designing data warehousing and data mining systems for real-world applications.

Course Content:

UNIT I

Introduction to data warehousing, Evolution of decision support systems, Modeling a data warehouse, granularity in the data warehouse, Data warehouse life cycle, building a data warehouse, Data Warehousing Components, Data Warehousing Architecture.

UNIT II

On Line Analytical Processing, Categorization of OLAP Tools, Introduction to Data mining and knowledge discovery, Relation to Statistics, Databases, Data Mining Functionalities, Steps In Data Mining Process, Architecture of a Typical Data Mining Systems, Classification of Data Mining Systems.

UNIT III

Overview of Data Mining Techniques, Data Preprocessing, Data Cleaning, Data Integration, Data Transformation and Data Reduction, Data Generalization and Summarization Based Characterization, Mining Association Rules In Large Databases.

UNIT IV

Classification and Prediction, Issues Regarding Classification and Prediction, Classification By Decision Tree Induction, Bayesian Classification, Other Classification Methods.

Prediction, Clusters Analysis, Types of Data In Cluster Analysis, Categorization of Major Clustering Methods, Partitioning methods, Hierarchical Methods.

UNIT V

Applications of Data Mining, Social Impacts of Data Mining, Case Studies, Mining WWW, Mining Text Database, Mining Spatial Databases.

Text Books:

1. Adriaans, ***" Data mining"***, Addison- Wesley, 1996.
2. Margaret Dunham, ***"Data Mining: Introductory and Advanced Topics"***, Published by Prentice Hall.
3. Weiss, SholomM.,***"Predictive data mining : a practical guide",*** Kaufmann Publishers, 1998.

Reference Books:

1. Pang-Ning Tan, Michael Steinback, Vipin Kumar, ***"Introduction to Data Mining"***, Pearson Education, 2008.
2. M.Humphires, M.Hawkins, ***"Data Warehousing: Architecture and Implementation"***, Pearson Education, 2009.
3. Anahory, Murray, ***"Data Warehousing in the Real World"***, Pearson Education, 2008. Kargupta, Joshi, etc.,
4. ***"Data Mining: Next Generation Challenges and Future Directions"***, Prentice Hall of India Pvt. Ltd, 2007.

Course Title:	**Compiler Design**	**Semester VI**	
Course Code	**BTITOE604A**	**Course Type**	**Elective**
Pre-requisite	**Data Structures & Applications**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To introduce the major concept areas of language translation and compiler design.
2. To develop an awareness of the function and complexity of modern compilers.
3. To provide practical, hands on experience in compiler design.

Course Outcomes:

After learning the course, the students should be able:

1. To understand the major concept areas of language translation and compiler design.
2. To develop an awareness of the function and complexity of compilers.
3. To identify the similarities and differences among various parsing techniques and grammar transformation techniques.

Course Content:

UNIT I

Introduction to Automata Theory and Context Free Grammar: Definition of deterministic finite automata, Non-deterministic finite automata, Regular expressions, Recursive definition, NFA with e-moves, Regular expression and FA, Production rules, Ambiguous grammar, Removal of ambiguity, Chomsky hierarchy, Context Free Grammar (CFG) – definition, Simplification of CFG. Pushdown Automata- Definition, Non-deterministic PDA, Multi-stack PDA, Definition:Turing Machines, TM construction.

UNIT II

Introduction to Compiling and Lexical Analysis: Definition, analysis of the source program, the phases of a compiler, the grouping of phases, Compiler-Construction tools, Role of the Lexical analyzer, Input buffering, Specification of Tokens, A Language for Specifying Lexical Analyzers, Design of a Lexical Analyzer generator.

UNIT III

Syntax Analysis: The role of the Parser, Context-free grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Operator-precedence Parsing, LR-Parsers, Using Ambiguous Grammars, Parser Generators.

UNIT IV

Syntax-Directed Translation: Definitions, Construction of Syntax Trees, Bottom-Up Evaluation of S Attributed definitions, Top-Down Translation, Bottom-Up Evaluation of Inherited attributes.

UNIT V

Intermediate Code Generation: Variants of syntax trees, Three address code, Types and declarations, Type checking, Control Flow and Backpatching Code Generation and Code Optimization: Code Generation: Issues in the design of code generation, The target language, Addresses in the target code, Different forms of object codes, Register allocation and assignments, Code Optimization:Need of code optimization, Principal sources of optimization, Optimization of basic blocks, Peephole optimization

Text Books:

1. Aho, Sethi, Ullman, ***"Compilers-Tools and Techniques"***, Pearson, 2nd Edition, 2011.
2. Tremblay, Sorenson, ***"Theory and Practice of Compiler Writing"***, McGraw Hill Publication.
3. Hopcroft, ***"Introduction to Automata Theory, Languages and Computation"***, Pearson Publication.

Reference Books:

1. Paul G. Sorenson, ***"Compiler Writing"***, Tata McGraw Hill.
2. Robin Hunter, ***"The Essence of Compilers"***, Pearson Publication, 1998.

Course Title:	**Enterprise Resource Planning**	**Semester VI**	
Course Code	**BTITOE604B**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To introduce to enterprise systems and show how organizations use enterprise systems to run their operations more efficiently and effectively.
2. To learn about the critical success factors and implementation strategies that lead to enterprise system success.
3. To learn about the informational, knowledge, and decision-making opportunities afforded by enterprise systems.
4. To examine typical Enterprise Systems modules: materials management (MM), supply chain management (SCM), customer relationship management (CRM), financials, projects, human resource management (HRM).

Course Outcomes:

After learning the course, the students should be able:

1. To demonstrate a good understanding of basic issues in Enterprise Systems.
2. To explain the scope of common Enterprise Systems (e.g., MM, SCM, CRM, HRM, procurement).
3. To explain the challenges associated with implementing enterprise systems and their impacts on organizations.
4. To describe the selection, acquisition and implementation of enterprise systems.
5. To use one of the popular ERP packages to support business operations and decision-making.
6. To communicate and assess an organization's readiness for enterprise system implementation with a professional approach in written form.
7. To demonstrate an ability to work independently and in a group.

Course Content:

UNIT I

Enterprise Resource Planning: Introduction, Disadvantages of non-ERP systems, What Is ERP? Need of ERP, Advantage of ERP, Risks of ERP, Growth of ERP.

ERP Modules: Finance, Production Planning, Control and Management, Sales and Distribution, Human Resource Management, Inventory Control System, Quality Management, Plant Maintenance.

UNIT II

ERP Implementation: ERP Implementation (Transition) strategies, ERP Implementation Life Cycle, Implementation Methodologies, Evaluation and selection of ERP package, ERP Project Team: Vendors, Employees, Consultants, Training & Education, Project management & Monitoring, Post Implementation Activities, Operation & maintenance of ERP system, Measuring the Performance of ERP System, Success & failure factors of an ERP, Implementation.

UNIT III

ERP Market and Vendors: ERP Marketplace and Marketplace Dynamics, Comparison of Current ERP Packages and Vendors, like; SAP, Oracle, PeopleSoft, BAAN etc.

UNIT IV

ERP and Related Technologies: Business Process Re-Engineering (BPR), Information Systems - Management Information, System (MIS), Decision Support System (DSS), Executive Support System (ESS) Data Warehousing, Data Mining, On-Line Analytical Processing (OLAP), Supply Chain Management, Customer Relationship Management.

UNIT V

ERP Case Studies: ERP systems implemented in: TISCO, SKF Automotive Bearings Co. Ltd, Qualcomm CDMA, California, Post Implementation review of ERP packages in Manufacturing, Services and Others Organizations, Customization of ERP for different types of Industries.

Text Books:

1. Alexis Leon, ***"ERP Demystified"***, TMH New Delhi, 2nd Edition.
2. V. K. Garg& N. K. Venkita Krishnan, ***"ERP Ware: ERP Implementation Framework"***, PHI.

Reference Book:

1. V. K. Garg& N. K. Venkita Krishna, ***"ERP Concepts & Planning"***, PHI, 2nd Edition.

Course Title:	**Decision Support Systems**	**Semester VI**	
Course Code	**BTITOE604C**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To select appropriate modeling techniques for supporting semi-structured business decision making.
2. To identify and select appropriate decision support systems for generating innovative business solutions.
3. To design and implement decision support systems for generating innovative business solutions.

Course Outcomes:

After learning the course, the students should be able:

1. To recognize the relationship between business information needs and decision making.
2. To appraise the general nature and range of decision support systems.
3. To appraise issues related to the development of DSS.
4. To select appropriate modeling techniques.
5. To analyze, design and implement a DSS.

Course Content:

UNIT I

Basic Concepts: Decision making systems, Modeling and support, Basics and definition Systems models, Modeling process, Decision making, Intelligence phase, Design phase Choice phase, Evaluation, Implementation phase, Alternative decision making models, Decision support systems, Decision makers, Case applications.

UNIT II

Decision Support System Development: Decision support system development, Basics, Life cycle, Methodologies, Prototype, Technology levels and tools, Development platforms, Tool selection, Developing DSS, Enterprise systems, Concepts and definition, Evolution of information systems, Information needs, Characteristics and capabilities, Comparing and integrating EIS and DSS, EIS data access, Data warehouse, OLAP, Multidimensional analysis, Presentation and the Web, Including soft information enterprise on systems, Organizational DSS, Supply and value chains, Decision support, Supply chain problems and solutions, Computerized systems. MRP, ERP, SCM, Frontline decision support systems.

UNIT III

Knowledge Management: Organizational learning and memory, Knowledge management, Development Methods, Technologies and tools, Success , Knowledge management and artificial

intelligence, Electronic Document Management, Knowledge Acquisition and Validation, Knowledge Engineering – Scope, Acquisition Methods, Interviews, Tracking Methods, Observation and other Methods, Grid Analysis, Machine Learning, Rule Induction, Case-Based Reasoning, Neural Computing, Intelligent Agents, Selection of an appropriate Knowledge Acquisition Methods, Multiple Experts, Validation and Verification of the Knowledge Base-Analysis, Coding, Documenting, and Diagramming, Numeric and Documented.

UNIT IV

Knowledge Acquisition, Knowledge Acquisition and the Internet/Intranets, Knowledge Representation Basics, Representation in Logic and other Schemas, Semantic Networks, Production Rules, Frames, Multiple Knowledge Representation, Experimental Knowledge Representations, Representing Uncertainty. Intelligent System Development: Inference Techniques, Reasoning in Artificial Intelligence, Inference with Rules, Inference Tree, Inference with Frames, Model Based and Case Based Reasoning, Explanation and Meta Knowledge, Inference with Uncertainty, Representing Uncertainty, Probabilities and Related Approaches, Theory of Certainty, Approximate Reasoning using Fuzzy Logic. Intelligent Systems Development, Prototyping, Project Initialization, System Analysis and Design, Software Classification.

UNIT V

Building Expert Systems with Tools, Shells and Environments, Software Selection, Hardware, Rapid Prototyping and a Demonstration Prototype, System Development, Implementation, Post Implementation. Management Support Systems: Implementing and Integrating Management Support Systems, Implementation, Major Issues, Strategies, System Integration, Generic Models MSS, DSS–ES, Integrating EIS, DSS and ES, Global Integration, Intelligent DSS, Intelligent Modeling and Model Management, Examples of Integrated Systems, Problems and Issues in Integration.

Text Book:

1. Efrain Turban and Jay E. Aronson, "***Decision Support Systems and Intelligent Systems"***, Pearson Education, 6th Edition, 2001.

Reference Books:

1. Ganesh Natarajan and SandhyaShekhar, ***"Knowledge Management Enabling Business Growth"***, Tata McGraw Hill, 2002.
2. George M. Marakas, ***"Decision Support System"***, Prentice Hall, India, 2003.
3. Efrem A. Mallach, ***"Decision Support and Data Warehouse Systems"***, Tata McGraw Hill,2002.
4. KimizDalkir, ***"Knowledge Management: Theory and Practice"***, Elsevier Science, 2005.
5. Becerra Fernandez and Laidener, ***"Knowledge Management: An Evolutionary View"***, PHI, 2009.

Course Title:	**Software Project Management**	**Semester VI**	
Course Code	**BTITOE604D**	**Course Type**	**Elective**
Pre-requisite	**Software Engineering**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To gain knowledge about the concepts and methods required for construction of large software intensive system.
2. To gain knowledge on the principles and techniques of software project managements.
3. To gain knowledge about organization behavior and general Management techniques used for project management.

Course Outcomes:

After learning the course, the students should be able:

1. To apply the process to be followed in the software development life-cycle models.
2. To understand approaches for managing and optimizing the software development process.
3. To explain the quality management and different types of metrics used in software development.
4. To do the Project scheduling, tracking, Risk Analysis, Quality Management and Project cost estimation using different techniques and tools.

UNIT I

Project Evaluation and Planning - Activities in Software Project Management, Overview of Project Planning, Stepwise planning, Software processes and process models, Cost Benefit Analysis, Cash Flow Forecasting, Cost-Benefit Evaluation Techniques, Risk Evaluation.
Software effort estimation, Activity Planning, Risk Management, Resourse Allocation.

UNIT II

Monitoring and Control- Collecting Data, Visualizing Progress, Cost Monitoring, review techniques, project termination review, Earned Value analysis, Change Control, Software Configuration Management (SCM).

UNIT III

Managing Contracts: Types of Contracts, Stages In Contract Placement, Typical Terms of A Contract, Contract Management and Acceptance.

UNIT IV

Quality Management and People Management- Introduction, Understanding Behavior, Organizational Behaviour, Selecting The Right Person for The Job, Motivation, The Oldman – Hackman Job Characteristics Model, working in Groups, Organization and team structures, Decision Making, Leadership, Organizational Structures, Stress, Health and Safety. ISO and CMMI models, Testing, and Software reliability, test automation.

UNIT V

Overview of project management tools.

Text Book:

1. Bob Hughes, Mike Cotterell, ***"Software Project Management"***, Tata McGraw Hill, 6th edition, 2017.

Reference Books:

1. Wakker Royce, ***"Software Project Management"***, Pearson Education, 2002.
2. Robert K. Wysocki, ***"Effective Software Project Management"***, Wiley, 2006.

Course Title:	**Database Management Systems Lab**	**Semester VI**	
Course Code	**BTITL605**	**Course Type**	**Compulsory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Creation of databases and use of SQL commands (DDL, DML and DCL).
2. Suitable exercises to practice SQL commands may be given for Insert, Update and Delete.
3. Write SQL procedure for an application which uses exception handling.
4. Write SQL procedure for an application with cursors.
5. Write SQL for implementing Nested Queries.
6. Write SQL for implementing Join Queries.
7. Write a DBMS program to prepare reports for an application using functions.
8. Write SQL block containing triggers.
9. Write SQL block containing stored procedures.
10. Develop a menu driven, GUI-based database application in any one of the domains such as Banking, Billing, Library management, Payroll, Insurance, Inventory, Healthcare etc. integrating all the features specified in the above exercises.

Course Title:	**Operating Systems Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Compulsory**
Prerequisite		**L – T – P**	**0– 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Basics of UNIX commands.
2. Shell Programming.
3. Implement the following CPU scheduling algorithms:
   Round Robin, SJF, FCFS, Priority scheduling
4. Implement all file allocation strategies:
   Sequential, Indexed, Linked
5. Implement Semaphores.
6. Implement all File Organization Techniques:
   Single level directory, Two level, Hierarchical, DAG
7. Implement Bankers Algorithm for Dead Lock Avoidance.
8. Implement an Algorithm for Dead Lock Detection.
9. Implement e all page replacement algorithms:
   FIFO, LRU, LFU
10. Implement Shared memory and IPC.
11. Implement Paging Technique of memory management.
12. Implement Threading & Synchronization Applications.

Course Title:	**Software Testing Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Elective**
Prerequisite	**Software Engineering**	**L – T – P**	**0– 0 – 2**
Stream	**Software application & Development**	**Credit**	**1**

Lab Experiments List:

1. Design, develop, code and run the program in any suitable language to solve the commission problem.Analyze it from the perspective of data flow testing, derive at least 10 different test cases, execute thesetest cases and discuss the test results.

2. Design, develop, code and run the program in any suitable language to solve the NextDate problem.Analyze it from the perspective of decision table-based testing, derive at least 10 different test cases,execute these test cases and discuss the test results.

3. Design, develop, code and run the program in any suitable object-oriented language to solve the calendarproblem. Analyze it from the perspective of OO testing, derive test cases to test the method that increment the date and the method that increments the month., execute these test cases and discuss the test results.

4. Design, develop, code and run the program in any suitable object-oriented language to solve the currencyconverter problem. Analyze it from the perspective of use case-based system testing, derive appropriatesystem test cases, execute these test cases and discuss the test results.

5. Design, develop, code and run the program in any suitable language to implement an absolute letter gradingprocedure, making suitable assumptions. Determine the basis paths and using them derive different testcases, execute these test cases and discuss the test results.

6. Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

Course Title:	**Data Storage Technologies & Networks Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Elective**
Pre-requisite	**Computer Networks & Internetworking Protocol, Operating Systems**	**L – T – P**	**0 – 0 – 2**
Stream	**Infrastructure & Security Management**	**Credit**	**1**

Lab Experiments List:

1. Install a hard disk on a Linux machine covering all the below activities:

   a) Connecting the disk to an HBA (Host Bus Adapter) and BIOS setup for the disk;
   b) Partitioning the disk;
   c) Creating file systems within disk partitions;
   d) Mounting the files systems;
   e) Setting up automatic mounting;
   f) Labeling disk partitions;
   g) Setting up swapping on swap partitions.

2. Use "smartmontools" to monitor the disk performance monitoring and testing:

   a) Use "smartctl" to enable S.M.A.R.T. support and offline data collection on the disk;
   b) Check the overall health of the disk;
   c) Run a self-test on the disk;
   d) Set up "smartd" to do tests automatically.

3. Use "hdparm", "iostat", and "iometer" tools to measure the performance of different storage devices, such as SATA drive, SCSI drive, and USB drives.

   a) Plot graphs to compare read/write and sequential/random access rates among different storage devices.

4. Use Navisphere Manager Simulator to perform management on SAN disk array systems:

   a) Configure storage pools and LUNs (Logical Unit Number) for storage groups;
   b) Configure snapshots and clones;
   c) Create SANCopy full and incremental sessions;
   d) Create MirrorView synchronous and asynchronous images;
   e) Expand a LUN to create metaLUNs;
   f) Migrate a LUN to another LUN.

5. Use Openfiler for network storage configuration management:

   a) Configure the Openfiler to support locally attached USB drives;
   b) Set up a NAS server to support NSF and CIFS protocols;
   c) Set up a SAN server to support an iSCSI protocol.

6. Configure Openfiler as a NAS Server:

   a) Configure access control rules and NFS/CIFS shares for the NAS server;

b) Configure the Linux client machine to access the NFS shares on the NAS server;
c) Configure a Windows VM on the Linux client machine to access the CIFS shares on the NAS server;
d) Use Openfiler to set up a SAN server, to supports iSCSI protocol for the block level data access;
e) Configure access control rules for the SAN server and configure iSCSI targets on the server.

7.

a) Use VMware to create virtual disks, Virtual Machine File Systems and provisioning.
b) Use thin and thick provisioning concepts.

Course Title:	**Service Oriented Architecture Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Compulsory**
Prerequisite		**L – T – P**	**0– 0 – 2**
Stream	**Information Management & Quality Control**	**Credit**	**1**

Lab Experiments List:

1. To create a web service for adding few numbers using NetBeans.
2. To create a web service for adding few numbers using NetBeans and write client-side code to invoke the web service.
3. Creation of A Web Service with Database Connectivity
4. Create a SOA project with BPEL Module to compose a web service
5. To develop a web service program which can persists the records of a student in the exam table. It makes use of SOAP Request and SOAP Response
6. To invoke EJB components as web services.
7. To create a web services in .NET.
8. To invoke j2ee web services from .net clients
9. To create components using .NET client.
10. To access .net web services from J2EE client.
11. Develop a Service Orchestration Engine (workflow) using WS-BPEL and implement service composition (Study Experiment)

Course Title:	**Network Programming Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Network**	**Credit**	**1**

Lab Experiments List:

1. Getting started with Basics of Network configurations files and Networking Commands in Linux.
2. To familiarize and understand the use and functioning of System Calls used for Operating system and network programming in Linux.
3. Familiarization and implementation of programs related to Process and thread.
4. Implement programs for Inter Process Communication using PIPE, Message Queue and Shared Memory.
5. Implement Client-Server communication using Socket Programming and TCP as transport layer protocol.
6. Implement Client-Server communication using Socket Programming and UDP as transport layer protocol.
7. Implement and simulate algorithm for Distance vector routing protocol.
8. Implement and simulate algorithm for Link state routing protocol.
9. Implement Simple Mail Transfer Protocol.
10. Using Wireshark observe data transferred in client server communication using UDP and identify the UDP datagram.
11. Using Wireshark observe Three Way Handshaking Connection Establishment, Data Transfer and Three Way Handshaking Connection Termination in client server communication using TCP.
12. Develop a packet capturing and filtering application using raw sockets.

Course Title:	**Data WareHousing & Data Mining Lab**	**Semester VI**	
Course Code	**BTITL606**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Data Science**	**Credit**	**1**

Lab Experiments List:

1. Build Data Warehouse/Daata Mart (using open source tools likee Pentaho Data Integration Tool, Pentaho BBusiness Analytics; or other data warehhouse tools like Microsoft-SSIS,Informatica,BBusiness Objects,etc.,)
2. Design multi-demesional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking,Insurance, Finance, Healthcare, manufacturing, Automobiles,sales etc).
3. Write ETL scripts and implement using data warehouse tools.
4. Perform Various OLAP operations such slice, dice, roll up, drill up and pivot.
5. Explore visualization features of the tool for analysis like identifying trends etc.
6. Explore WEKA Data Mining/Machine Learning Toolkit
7. Load each dataset into Weka and run Aprior algorithm with different support and confidence values. Study the rules generated.
8. Apply different discretization filters on numerical attributes and run the Aprior association rule algorithm. Study the rules generated. Derive interesting insights and observe the effect of discretization in the rule generation process.
9. Load each dataset into Weka and run id3, j48 classification algorithm, study the classifier output. Compute entropy values, Kappa ststistic.
10. Extract if-then rues from decision tree gentrated by classifier, Observe the confusion matrix and derive Accuracy, F- measure, TPrate, FPrate , Precision and recall values. Apply cross-validation strategy with various fold levels and compare the accuracy results.
11. Load each dataset into Weka and run simple k-means clustering algorithm with different values of k(number of desired clusters). Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.

Course Title:	**Database Programming**	**Semester V**	
Course Code	**BTITAO01**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Course Objective:

1. Introduction to Relational Database Management
2. To learn what is Entity-Relationship Diagrams and significance of it
3. To learn Normalization and its usages in real world. Impact of denormalization.
4. To learn SQL for Data Retrieval and Manipulation and explore its usages to meet the business need
5. To learn how to manage data integrity using Constraints
6. Query Optimization using indexes
7. Develop efficient PL/SQL programs for data manipulation
8. Design Modular application using packages
9. Invoke Native dynamic SQL to build runtime SQL statements
10. Learn advance programming techniques such as cursors and cursor variables
11. Enhance performance using collection datatypes and bulk operations
12. Create database triggers to solve business challenges and to enforce business rules
13. To manage various database objects and dependencies

Course Outcome:

After completing this course students would be able:

1. To understand the importance of Database Management System in real world
2. To design Database and required components as per business need by using the knowledge of Normalizations, Entity-Relationship, Constraints, Procedures, Functions, Packages, Triggers, Views etc.
3. To manipulate data using SQL (DML/DDL) statements, procedures, functions, packages, views, triggers etc.
4. To understand performance issues and its resolution.

Course Content:

UNIT I

Overview of Database Management system and Programming., Database models: Flat, Hierarchical, Network, Relational., Entity-Relationship Diagrams.

Relational database design. Types of database design.

Relationships: One-to-One, One-to-Many, Many-to-Many.

Normalization: First, Second, Third and Higher Normal Forms, Denormalization.

Integrity rules. What is Data, Data Definition: Schemas and User, Data Dictionary.

Schemas, Types of Schema. Database Users.

User Management- Database User creation, Privileges, Roles and Roles Management.

UNIT II

DDL statements for Table, Object Types, Users and Schema, Object Namespaces

Understanding table design. List of various datatypes available for Columns of the table. Create Table, Alter Table, Dropping and Truncating tables.

Constraints, Types of Constraints, Constraint Creation. Addition of Constraints. View Constraints Information, Constraint Management (Enable/Disable, Dropping).

Data Retrieval using Select statement. Usages of Arithmetic Operators. SQL Expressions and Operators.

Data Restricting and Sorting: Limit the rows, Comparison Operators. LIKE operator, Boolean Operator, NULL values, Precedence Rule. Data sorting. Substitution variables.

Single Row Functions: Defining a Function, Types of Function, Use of Character, Number and Date Functions.

Conversion Functions and Conditional Expressions. TO_CHAR, TO_DATE, TO_NUMBER. Apply Conditional Expressions like NULLFIF, NVL2, COALESCE, DECODE, CASE expression.

UNIT III

Reporting Aggregated Data using Group Functions. Describe and Identify Group Functions. Data Grouping Restricting using GROUP BY and HAVING clause.

Joins: Equality Join, Non-Equality Join, Outer Joins, Cartesian Join, Self-Join.

Set operators and types, Sets and Venn Diagrams.

Define Subqueries. Types of Subqueries: Single -row, Multiple-row, Co-related and Nested subqueries.

Data Manipulation Overview and Commands (DML). INSERT, UPDATE, DELETE, MERGE, TRUNCATE. Insert, update and delete rows from the table.

What is Transaction. Transaction Controlling using COMMIT, ROLLBACK and SAVEPOINT. Lock Table, SELECT …FOR UPDATE command.

View Introduction, Simple and Complex view, View creation. Data Manipulation using View, Dropping View, Materialized View.

Other database objects like Sequences, Indexes and Synonyms.

UNIT IV

PL/SQL Programming Language Overview. PL/SQL processing Architecture, Two-Tire, N-Tire Model, PL/SQL Block Structure, Types of PL/SQL blocks.

Delimiters, Identifiers, Literals, Comments, Variable and Data types

Control Structure: Conditional Statements (IF statements, CASE statements, GOTO and NULL statements),

Iterative Processing Statements: Simple Loop, WHILE Loop, FOR Loop,

Cursor Structures: Implicit Cursors, Explicit Cursors, Cursor Attributes, Cursor FOR LOOP, Parametric Cursor, FOR UPDATE and WHERE CURRENT Clauses.

Bulk Statements: BULK COLLECT INTO, FOR ALL

Dynamic SQL: Architecture, Native Dynamic SQL, DBMS_SQL package

Object Types: Declare, Implement, White Listing, Getter and Setter, Methods, Inheritance and Polymorphism, Implement Object Type collection.

Triggers: Trigger Architecture, DDL , DML Triggers, Instead of Triggers, System and Database Event Triggers. Trigger Restrictions: Trigger size, SQL statements, Long Data Types, Mutating Tables. Managing Triggers (Enable, Disable, Removal).

UNIT V

Collections Introductions. Object Types: Varray Collection, Table Collections, Collection Methods

Error Management - Exception Types, Compilation Error, Runtime Error, Exception Management using Built-in Functions, User Defined Exceptions.

Functions and Procedures. Calling Subroutines, Positional Notation, Named Notation, Mixed Notation, SQL Call Notation. Function and Procedure Creation, Pass-by-Value, Pass-by-Reference.

Packages – Package Architecture. Package Specification: Variables, Types, Components (Function and Procedure), Serially Reusable Precompiler Directive. Package Body: Variables, Types, Components (Procedure and Function). Definer vs. Invoker Rights, Managing Packages in Database: Finding, Validating and Describing, Checking Dependencies, Overloading subroutines.

Transaction Management: COMMIT, ROLLBACK, SAVEPOINT. LOCKING, Transaction Isolation, Deadlocks.

Large Objects: Introduction, LOB Types, SQL to work with LOBs (CLOB, NCLOB, BLOB, BFILE), Use of PL/SQL to work with LOBs.

SQL Tuning: EXPLAIN PLAN statement, DBMS_XPLAN package.

SQL Tracing: Tracing Session Statement, Convert Raw Trace File to Readable format

Text Books

1. OCA Oracle Database 12c SQL Fundamentals I Exam Guide (Exam 1Z0-061), 2nd Edition by RoopeshRamklass, Published by Oracle Press, 2014
2. Oracle database 12c PL/SQL Programming by Michael McLaughlin, McGraw Hill Education, 2014.

Reference Books

1. J. Murach, “Murach’s Oracle SQL and PL/SQL: Works with All Versions Through 11g”, Shroff Publication, 2008.
2. Oracle Database SQL Language Reference, 12c Release 2 (12.2), part of Oracle Documentation.
3. Oracle Database PL/SQL Language Reference, 12c Release 2 (12.2), part of Oracle Documentation.

Course Title:	**Deep Neural Networks**	**Semester V**	
Course Code	**BTITAO02**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus prepation is in progress.

Course Title:	**Compliance, risk and Governance: IT Perspective**	**Semester V**	
Course Code	**BTITAO03**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Course Objective:

1. To know the goals and objectives of IT GRC.
2. Deeper dive into concepts, tools and applications for each of areas within GRC.
3. Introduction to key processes, tools, methodologies, and framework – as applicable for GRC.
4. To learn the fundamentals of SOX.

Course Outcome:

After completing this course students would be able:

1. To describe the need for IT GRC.
2. To be aware of the basic building blocks of GRC and apply the learned knowledge as well relate to real scenarios in IT industry with respect to GRC.
3. To explain the mandatory items that need to be checked.
4. Familiarize themselves with actual SOX controls, TOD and TOE as used in IT industry.

UNIT I

IT Governance: Introduction, fundamentals, why and how. Role of IT Governance.Peek into COBIT, ITIL, COSO, CMMI, FAIR.

UNIT II

IT Risk: Risk Response, Risk Identification, Assessment and Evaluation, Risk Response, Risk Monitoring, Information Systems Control Design and Implementation.

UNIT III

Information Systems Control Monitoring and Maintenance.

UNIT IV

IT Compliance: Introduction, Internal and External Compliance, Regulatory Compliance Standards.

UNIT V

IT Compliance Goals and Challenges, Peek into SOX.GLBA, FISMA, HIPAA, PCI DSS, SSAE 16, Basel III. The Sarbanes -Oxley Act (SOX) : Deeper dive in context of IT, Control areas, List of actual controls, applicability, Test of Design (TOD), Test of Execution (TOE).

Text Books:

1. To be confirmed.

Reference Books:

1. To be confirmed.

Course Title:	**ERP: Oracle Application**	**Semester V**	
Course Code	**BTITAO04**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Database Programming Lab**	**Semester V**	
Course Code	**BTITAOL01**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Deep Neural Network Lab**	**Semester V**	
Course Code	**BTITAOL02**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Compliance, risk and Governance: IT Perspective Lab**	**Semester V**	
Course Code	**BTITAOL03**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**ERP: Oracle Application Lab**	**Semester V**	
Course Code	**BTITAOL04**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Machine Learning**	**Semester VII**	
Course Code	**BTITC701**	**Course Type**	**Compulsory**
Pre-requisite	**Engineering Mathematics III**	**L – T – P**	**3 – 1 – 0**
Stream	**Core**	**Credits**	**4**

Course Objectives:

1. To understand the basic concepts and methods of machine learning.
2. To make use of some elementary machine learning techniques in the design of computer systems.
3. To develop a broad perspective about the applicability of ML algorithms in different fields.
4. To understand the major machine learning algorithms, the problem settings and assumptions that underlies them.
5. To possess insights, concerning the relative strengths and weaknesses of various common machine learning methods.

Course Outcomes:

After learning the course the student will be able:

1. To demonstrate knowledge of the machine learning literature.
2. To describe how and why machine learning methods work.
3. To demonstrate results of parameter selection.
4. To explain relative strengths and weaknesses of different machine learning methods.
5. To select and apply appropriate machine learning methods to a selected problem.
6. To implement machine learning algorithms on real datasets.
7. To suggest ways to improve results.

Course Content:

UNIT-I

Introduction: Well-posed learning problems, Designing a Learning System, Perspectives and Issues in Machine learning, Concept Learning and General-to-specific Ordering: A concept learning task, Concept learning as Search, Finding a maximally specific hypothesis, Version Spaces and Candidate elimination algorithm, Inductive Bias.

UNIT-II

Decision Tree Learning: Decision tree learning algorithm, Hypothesis space search in decision tree Evaluating Hypothesis: Estimating Hypothesis accuracy, Basics of sampling theory, Deriving confidence intervals, Hypothesis testing, comparing learning algorithms.

UNIT-III

Bayesian Learning: Bayes theorem and concept learning, Maximum likelihood and least square error hypotheses, Minimum description length principle, Bayes optimal classifier, Gibbs algorithm, Naive Bayes classifier, Computational Learning Theory: Probably learning an approximately correct hypothesis, PAC learnability, The VC dimension, the mistake bound model for learning.

UNIT-IV

Models for Regression: Linear basis function models, The Bias-Variance decomposition, Bayesian Linear Regression, Logistic regression, SVM.

Kernel Methods: Constructing kernels, Radial basis function networks, Gaussian Processes

UNIT-V

Hidden Markov Models: Learning algorithms for HMM, the Viterbi algorithm, Linear Dynamical Systems. Clustering: k-means, adaptive hierarchical clustering, Gaussian mixture model.

Ensemble Learning, Reinforcement Learning: The learning task, Q learning, Non-deterministic rewards and action, Temporal difference learning, Generalizing from examples.

Text Book:

1. Mitchell Tom. M., *"**Machine Learning**"*, McGraw-Hill Education, 1st Edition, May 2013.

Reference Books:

1. Miroslav, Kubat. ***"An Introduction to Machine Learning"***, Springer Publishing.
2. Bishop, C. M., ***"Pattern Recognition and Machine Learning"***, Springer Publishing.

Course Title:	**Pattern Recognition**	**Semester VII**	
Course Code	**BTITPE702A**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Professional Elective**	**Credits**	**3**

Course Objectives:

1. To study pattern recognition topics and be exposed to recent developments in pattern recognition research.
2. To provide in-depth design concepts and implementation techniques of pattern recognitions.

Course Outcomes:

1. Identify and explain detailed aspects of internal structures of pattern recognitions.
2. Compare and contrast design issues for statistical pattern recognition.
3. Develop implementation skills for building pattern recognition.

Course Content:

UNIT I

Introduction: Machine Perception, Definition of Pattern Recognition (PR), Pattern Recognition system: Sensing, Segmentation & grouping, Feature extraction, Classification and Post processing, Design cycle: Data collection, Feature choice, Model choice, Training, Evaluation and computational complexity. Learning and adaptation: Supervised learning, Unsupervised learning and Reinforcement learning. Examples of PR Applications, Pattern Recognition Extensions. Machine learning : Components of learning, Learning models, Geometric models, Probabilistic models, Logic models, Grouping and grading, Learning versus design, Theory of learning, Feasibility of learning, Error and noise, Training versus testing, Theory of generalization, Generalization bound, Approximation-generalization tradeoff, Bias and variance, Learning curve.

UNIT II

Statistical Pattern Recognition (StatPR): Introduction to StatPR, Baye's theorem, Multiple features, Conditionally independent features, Decision boundaries, Unequal costs of error, Estimation of error rates, Characteristic curves, Estimating the composition of populations, Introduction to supervised parametric approaches and unsupervised approaches. Cluster analysis: Clustering techniques, Cluster analysis, Cluster validity. Feature selection & extraction: Feature selection criteria, Feature set search algorithm, Feature selection.

UNIT III

Tree Classifiers: (a) Decision Trees: CART, C4.5, ID3, (b) Random Forests, Linear Discriminants, Discriminative Classifiers: the Decision Boundary, (a) Separability, (b) Perceptrons, (c) Support Vector Machines.

Unsupervised Methods : Exploring the Data for Latent Structure :(a) Component Analysis and Dimension Reduction: i. The Curse of Dimensionality, ii. Principal Component Analysis, iii. Fisher Linear Discriminant, iv. Locally Linear Embedding

UNIT IV

Parametric Techniques: Generative methods grounded in Bayesian Decision Theory (a) Maximum Likelihood Estimation (b) Bayesian Parameter Estimation (c) Sufficient Statistics. Non-Parametric Techniques :(a) Kernel Density Estimators (b) Parzen Window (c) Nearest Neighbor Methods.

Clustering: i. K-Means, ii. Expectation Maximization, iii. Mean Shift. Classifier Ensembles : (a) Bagging, (b) Boosting / AdaBoost, Algorithm Independent, Topics Theoretical Treatments in the Context of Learned Tools: (a) No Free Lunch Theorem, (b) Ugly Duckling Theorem, (c) Bias-Variance Dilemma, (d) Jacknife and Bootstrap Methods.

UNIT V

Syntactic (Structural) Pattern Recognition (Syntpr): Introduction to SyntPR, Syntactic PR: primitive selection & pattern grammars, Higher dimensional grammars, Syntactic recognition, Automata, Error – correcting parsing, Shape & texture analysis, Image database management. Structural analysis using constraint satisfaction and structural matching, The Formal Language-based approach to SyntPR, Learning/Training in the Language-based Approach (Grammatical Inference). Problem solving methods for PR: Problem solving models, Problem solving algorithms.

Text Books:

1. Duda, R.O., Hart, P.E., Stork, D.G. ***"Pattern Classification",*** Wiley, 2nd Edition, 2001.
2. EartGose, Richard Johnsonburg and Steve Joust, ***"Pattern Recognition and Image Analysis"***, Prentice-Hall of India-2003.

Reference Books:

1. Bishop, C. M. ***"Pattern Recognition and Machine Learning"*** Springer, 2nd Edition, 2007.
2. Marsland, S., ***"Machine Learning: An Algorithmic Perspective"***, CRC Press. 2009.
3. Theodoridis, S. and Koutroumbas, K., ***"Pattern Recognition"***, 4th Edition, Academic Press, 2008.
4. Russell, S. and Norvig, N., ***"Artificial Intelligence: A Modern Approach",*** Prentice Hall, Series in Artificial Intelligence, 2003.

Course Title:	**Soft Computing**	**Semester VII**	
Course Code	**BTITPE702B**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Professional Elective**	**Credits**	**3**

Course Objectives:

1. To introduce a relatively new computing paradigm for creating intelligent machines useful for solving complex real world problems.
2. To gain insight into the tools that make up the soft computing technique: fuzzy logic, artificial neural networks and hybrid systems.
3. To create awareness of the application areas of soft computing technique.
4. To learn alternative solutions to the conventional problem solving techniques in image/signal processing, pattern recognition/classification, control system.

Course Outcomes:

After learning the course the student will be able:

1. To use a new tool /tools to solve a wide variety of real world problems.
2. To find an alternate solution, more adaptable, resilient and optimum.
3. To apply knowledge of soft computing domain to real world problems**.**

Course Content:

UNIT I

Artificial Neural Network: Biological neuron, Artificial neuron model, Concept of bias and threshold, McCulloch Pits Neuron Model, Implementation of logical AND, OR, XOR functions. Soft Topologies of neural networks, Learning paradigms: Supervised, Unsupervised, Reinforcement, Linear neuron model: Concept of error energy, Gradient descent algorithm and application of linear neuron for linear regression, Activation functions: Binary, Bipolar (linear, signup, log sigmoid, tan sigmoid) Learning mechanisms: Hebbian, Delta Rule of Perceptron and its limitations.

UNIT II

Artificial Neural Network: Multilayer perceptron (MLP) and back propagation algorithm, Application of MLP for classification and regression of self organizing Feature Maps, Clustering of Learning vector quantization. Radial Basis Function networks: Cover's theorem,Mapping functions (Gaussian, Multi-quadrics, Inverse multiquadrics, Application of RBFN for classification and regression of Hopfield network, Associative memories.

UNIT III

Fuzzy Logic: Concept of Fuzzy number, Fuzzy set theory (continuous, discrete) of operations on fuzzy sets, Fuzzy membership functions (core, boundary, support), Primary and composite linguistic terms, Concept of fuzzy relation, Composition operation (T-norm, T-conorm) of Fuzzy if-then rules.

UNIT IV

Fuzzy Logic:Fuzzification, Membership value assignment techniques, De-fuzzification (Maxmembership principle, Centroid method, Weighted average method), Concept of fuzzy inference, Implication rules: Dienes-Rescher Implication, Mamdani Implication, Zadeh Implication, Fuzzy Inference systems: Mamdani fuzzy model, Sugeno fuzzy model, Tsukamoto fuzzy model, Implementation of a simple two-input single output FIS employing Mamdani model Computing.

UNIT V

Fuzzy Control Systems: Control system design, Control (Decision) Surface, Assumptions in a Fuzzy Control System Design, Fuzzy Logic Controllers, Comparison with traditional PID control, Advantages of FLC, Architecture of a FLC: Mamdani Type, Example Aircraft landing control problem.

Adaptive Neuro-Fuzzy Inference Systems (ANFIS): ANFIS architecture, Hybrid Learning Algorithm, Advantages and Limitations of ANFIS Application of ANFIS/CANFIS for regression.

Text Books:

1. LaureneFausett, ***"Fundamentals of Neural Networks: Architectures, Algorithms And Applications"***, Pearson Education, 2008.
2. Timothy Ross, "***Fuzzy Logic With Engineering Applications"***, 3rd Edition, John Wiley & Sons, 2010.
3. J.S. Jang, C.T. Sun, E. Mizutani, ***"Neuro- Fuzzy and Soft Computing"***, PHI Learning Private Limited.
4. S. N. Sivanandam, S. N. Deepa, ***"Principles of Soft Computing"***, John Wiley & Sons, 2007.

Reference Books:

1. John Hertz, Anders Krogh, Richard Palmer, ***"Introduction to the theory of neural computation"***, Addison –Wesley Publishing Company, 1991.
2. Simon Haykin, ***"Neural Networks A comprehensive foundation"***, Prentice Hall International Inc-1999.
3. José C. Principe Neil R. Euliano , W. Curt Lefebvre, ***"Neural and Adaptive Systems: Fundamentals through Simulations"***, John-Wiley & Sons, 2000.
4. Peter E. Hart, David G. Stork Richard O. Duda, ***"Pattern Classification"***, 2nd Edition, 2000.
5. SergiosTheodoridis ,KonstantinosKoutroumbas, ***"Pattern Recognition"***, 4th Edition, Academic Press, 2008.
6. Hung T. Nguyen, Elbert A. Walker, ***"A First Course in Fuzzy Logic"***, 3rd Edition, Taylor & Francis Group, LLC, 2008.
7. S. N. Sivanandam , S. Sumathi, S. N. Deepa, ***"Introduction to Fuzzy Logic using MATLAB"***, Springer Verlag, 2007.

Course Title:	**Artificial Neural Networks**	**Semester VII**	
Course Code	**BTITPE702C**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Professional Elective**	**Credits**	**3**

Course Objectives:

1. To understand the biological neural network and to model equivalent neuron models.
2. To understand the architecture, learning algorithm and issues of various feed forward and feedback neural networks.
3. To create awareness of the application areas of neural network and its model.

Course Outcomes:

By completing this course, the student will be able to:

1. Understand the role of neural network in engineering, artificial intelligence and cognitive Modeling
2. Understand the concepts and techniques of neural networks through the study of most important neural network models.
3. Evaluate whether neural network is appropriate to a particular application.
4. Apply neural networks to particular application and to know what steps to take.

Course Content:

UNIT I

Introduction: Biological Neuron – Artificial Neural Model - Types of activation functions – Architecture: Feedforward and Feedback, Convex Sets, Convex Hull and Linear Separability, Non-Linear Separable Problem. XOR Problem, Multilayer Networks.

Learning: Learning Algorithms, Error correction and Gradient Descent Rules, Learning objective of TLNs, Perceptron Learning Algorithm, Perceptron Convergence Theorem.

UNIT II

Supervised Learning: Perceptron learning and Non-Separable sets, α-Least Mean Square Learning, MSE Error surface, Steepest Descent Search, μ-LMS approximate to gradient descent, Application of LMS to Noise Cancelling, Multi-layered Network Architecture, Backpropagation Learning Algorithm, Practical consideration of BP algorithm.

UNIT III

Support Vector Machines and Radial Basis Function: Learning from Examples, Statistical Learning Theory, Support Vector Machines, SVM application to Image Classification, Radial Basis Function Regularization theory, Generalized RBF Networks, Learning in RBFNs, RBF application to face recognition.

UNIT IV

Attractor Neural Networks: Associative Learning Attractor Associative Memory, Linear Associative memory, Hopfield Network, application of Hopfield Network, Brain State in a Box neural Network, Simulated Annealing, Boltzmann Machine, Bidirectional Associative Memory, Application of Neural Algorithms & System: Linear Programming Modeling Networks, Character Recognition networks, Neural network control applications, networks for robot kinematics, Connectionist expert system for medical diagnosis.

UNIT V

Self-organization Feature Map: Maximal Eigenvector Filtering, Extracting Principal Components, Generalized Learning Laws, Vector Quantization, Self-organization Feature Maps, Application of SOM, Growing Neural Gas.

Text Books:

1. J. M. Zurada, "Introduction to Artificial Neural System", Jaico Publication 1994.
2. Satish Kumar, ***"Neural Networks: a classroom Approach"***, McGraw Hill Education (India) Pvt. Ltd, Second Edition.

Reference Books:

1. B. Yegnanarayana – "***Artificial neural network PHI Publication***", New Delhi 1998.
2. Stuart Russell, Peter Norving-" ***Artificial Intelligence A Modern Approach***", Second Edition Prentice Hall of India Private Limited New Delhi-110001, (2003).
3. E. Rich & K. Knight ***"Artificial Intelligence"*** (second edition),(McGraw Hill, 1991)
4. D. W. Patterson ***"Introduction to Artificial Intelligence***", (Prentice Hall, 1990)

Course Title:	**Real Time Systems**	**Semester VII**	
Course Code	**BTITPE703A**	**Course Type**	**Elective**
Pre-requisite	**Operating Systems, Design and Analysis of Algorithms**	**L – T – P**	**3 – 0 – 0**
Stream	**Software Application and Development**	**Credits**	**3**

Course Objectives:

1. To introduce students to the fundamental problems, concepts and approaches in the design and analysis of real-time systems.
2. To study issues related to the design and analysis of systems with real-time constraints. \
3. To learn real-time scheduling and schedulability analysis.
4. To understand formal specification and verification of timing constraints and properties.
5. To design methods for real-time systems.
6. To learn new techniques of state-of-the-art real-time systems research.

Course Outcomes:

After learning the course the student will be able:

1. To characterize real-time systems and describe their functions.
2. To analyze, design and implement a real-time system.
3. To apply formal methods to the analysis and design of real-time systems.
4. To apply formal methods for scheduling real-time systems.
5. To characterize and debug a real-time system.

Course Content:

UNIT I

Introduction: Hard vs. Soft real time systems, A reference model of real time system. Real-time scheduling: Clock driven approach, Weighted Round-robin approach, Priority driven approach, Dynamic vs. static system, Effective Release Times and Deadlines, EDF and LST algorithm, Optimality and Non-Optimality of the EDF and LST algorithms, Off line vs. online Scheduling.

UNIT II

Clock-Driven Scheduling: Static, Time-Driven scheduler, General structure of Cyclic Schedules, Cyclic Executives, Improving the Average Response Time of a-periodic Jobs, Scheduling Sporadic Jobs, Multiprocessor scheduling, Resource Access Control and Synchronization: Model of multiprocessor & distributed systems, task assignment, multiprocessor Priority-ceiling protocol, Elements of Scheduling Algorithms for End-to-End Periodic Tasks- IPS protocols, PM protocols, MPM protocol.

UNIT III

Priority Driven Scheduling of Periodic Tasks: Fixed priority vs. Dynamic priority algorithms, Maximum Schedulable Utilization, Optimality of the RM and DM algorithms, A Schedulability test for fixed-priority tasks with short response times, Sufficient Schedulability conditions for the RM and DM algorithm.

UNIT IV

Scheduling Aperiodic and Sporadic Jobs in Priority-Driven Systems: Assumptions and Approaches, Deferrable Servers, Sporadic Servers, Constant Utilization, Total Bandwidth and Weighted Fair-Queuing Servers.

UNIT V

Resources and Resource Access control: Resource contention, Resource access control, Nonpreemptive critical section, Basic Priority-Inheritance protocol, Basic Priority Ceiling Protocol, Stack based, Priority-ceiling protocol, preemption ceiling protocol.

Text Books:

1. Jane W. S. Liu, ***"Real-Time System"***, Pearson Education.
2. C. M. Krishna and K. G. Shin, ***"Real-Time Systems"***, McGraw Hill.

Reference Books:

1. Laplante, ***"Real Time System Design and Analysis: An Engineer Handbook"***, PHI.
2. Dr. K. V. K. Prasad, ***"Embedded Real Time System Concept Design and Programming"***, Wiley India.

Course Title:	**Information Security**	**Semester VII**	
Course Code	**BTITPE703B**	**Course Type**	**Elective**
Pre-requisite	**Computer Networks & Internetworking Protocols**	**L – T – P**	**3 – 0 – 0**
Stream	**Professional elective**	**Credits**	**3**

Course Objectives:

1. To understand information security's importance in the increasingly computer-driven world.
2. To master the key concepts of information security and its working.
3. To develop a security mindset.
4. To learn to critically analyze situations of computer and network security usage.

Course Outcomes:

After learning the course the student will be able:

1. To explain the challenges and scope of information security.
2. To explain security concepts as confidentiality, integrity and availability.
3. To explain the importance of cryptographic algorithms used in information security.
4. To identify and explain symmetric algorithms for encryption-based security of information.
5. To describe the access control mechanism used for user authentication and authorization.
6. To explain the use of security tools as firewalls and intrusion prevention systems.

Course Content:

UNIT I

Introduction to Information Systems: Security concepts, Computer security concepts, Threats, Attacks and assets, Security functional requirements, A security architecture for Open Systems, Computer security trends, Computer security strategy.

UNIT II

Cryptographic Tools: Confidentiality with symmetric Encryption, Message authentication and hash functions, Public-Key encryption, Digital signatures and key management, Practical Application: Encryption of stored data.

UNIT III

Models, Frameworks, Standards & Legal Framework: A structure and framework of compressive security policy, Policy infrastructure, Policy design life cycle and design processes, PDCA model, Security policy standards and practices - ISO 27001, SSE-CMM, IA-CMM, ITIL & BS 15000, BS7799, Understanding laws for information security: Legislative solutions, Contractual solutions, Evidential issues, International activity, Indian IT Act, Laws of IPR, Indian Copyright Act.

UNIT IV

Virus and Malware: Introduction & types of Malicious Software (Malware), Propagation-Infected Content-Viruses, Propagation-Vulnerability Exploit-Worms, Propagation-Social Engineering-SPAM E-mail, Trojans, Payload-System Corruption, Payload-Attack, Agent-Zombie, Bots, Payload-Information Theft-Keyloggers, Phishing, Spyware, Payload-Stealthing-Backdoors, Rootkits, Countermeasures, Controls: Access control principles, Subjects, Objects and access rights, Discretionary access control, Role-based access control.

UNIT V

Security issues: Database security challenge in the modern world, Federated databases, Securing mobile databases, Network security, Trusted and untrusted networks, Network attacks, Network security dimensions, Network attack- the stages; Using firewalls effectively; Privacy- Privacy invasion due to direct marketing, Outsourcing using data masking, Privacy issues in smart card applications, Ethical Hacking, Role of Cryptography in information security, Digital signatures.

Text Books:

1. Nina Gobole, ***"Information Systems Security: Security Management, Metrics, Framework and Best Practices"***, Wiley, 2008.
2. Mark Rhodes-Ousley, ***"Information Security: The Complete Reference"***, McGraw-Hill Education, 2nd edition, 2013.
3. Dhiren R. Patel, ***"Information Security Theory and Practice"***, PHI Learning, 2008.
4. Mark Stamp, ***"Information Security: Principles and Practice"***, Wiley, 2nd edition, 2011.

Reference Books:

1. Gary R. McGraw, ***"Software Security: Building Security In"*** Addison Wesley, 2006.
2. Ankit Fadia, ***"Network Security: A Hacker's Perspective"***, Course Technology Inc, 2nd edition, 2006.

Course Title: Electronic Payment Systems		**Semester VII**	
Course Code: BTITPE703C		**Course Type**	**Elective**
Pre-requisite: Nil		**L – T – P**	**3 – 0 – 0**
Stream:	**Professional Stream**	**Credits**	**3**

Course Objectives:

1. To understand common payment methods, working of different payment systems.
2. To learn basic payments processes and systems.
3. To understand emerging payments trend.
4. To gain knowledge on the underlying technologies governing payment systems.

Course Outcomes:

After learning the course, the students should be able:

1. To learn and speak Financial Services language.
2. To familiarize with banking regulations in the payment industry
3. Gain domain knowledge for career in financial industry: Banks, Insurance & NBFC

Course Content:

UNIT I

Evolution of payment systems in the digital world: Role of RBI in payment/clearing/settlement Indian payment systems: IMPS, NEFT/RTGS, eWallet, eKYC, AADHAR / AADHAR VAULT, RUPAY Debit/Credit cards, *99#, NACH,ABPS, BHIM, BHARAT PAY, CREDIT CARD, VISA/MASTER ROLE in CREDIT CARD PAYMENTS, CTS, UPI, BBPS, ATM. Transformation in Social media channels & Payments: ChatBot, WhatsApp, FB.

UNIT II

Risks in Payment Systems: Credit Risk, Liquidity Risk, Systemic Risk, Operational Risk. Risk mitigation techniques: Carefully chosen members, Novation, Central counterparty system, Loss sharing arrangements, Collateral, Other mitigation techniques like RECO. Relationship structures: Correspondent banking, Bilateral clearing, arrangements, Network managed banking.

UNIT III

Payment types: Book payments, Local payments, Domestic payments, Cross border payments.

Regional payments systems: USA payment systems :Fedwire, CHIPS, NSS, ACH, SEPA payment systems : TARGET2, STEP 2 (SCT/SDD) PE- AC, China payment system : CDFCPS/CIPS, Hong Kong payment system : CHATS Canadian payment system: LVTS, Indian payment systems: RTGS, NEFT, IMPS, UPI.

UNIT IV

Overview of SWIFT messaging: MT and MX messages, Role of SWIFT in payment systems, SWIFTnet Fin, File act, Interact, Browse SWIFT payment message processing – MT 1XX, MT 2XX, MT 9XX, MX PAIN/PACS, SWIFT Payment Messages examples, SWIFT for corporate.

UNIT V

Use of code in payment systems: Codes – IBAN, BBAN, BIC, BEI, UID, UPIC, ABA routing codes etc, IFSC, Foreign exchange transactions: Cash, TOM, Spot, Forwards, Interbank transactions, Merchant transactions, Exchange rate determination and rate computation, Cash management products: Concept of float, Cash concentration, notional pooling and sweep, Virtual account management (VAM), ACH filter/ACH block, Lockboxes. Impact of regulation: Basel, FATF/OFAC compliance, FATCA compliance, AML compliance, FRM compliance.
Practical: Working of ATMs, Insides of an ATM, Vulnerability Points, Care to be taken while using ATM

.

Text Book:

1. S. K. Nippani, B. K. Murthy, ***"Digital India Governance Transformation"***, 2018.

Online Reference for books & documentations:

https://rbidocs.rbi.org.in/rdocs/

Course Title:	**Distributed Computing**	**Semester VII**	
Course Code	**BTITPE703D**	**Course Type**	**Elective**
Pre-requisite	**Operating Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Networking**	**Credits**	**3**

Course Objectives:

1. To understand the major tools and techniques that allow programmers to effectively program the parts of the code that require substantial communication and synchronization.
2. To study the core ideas behind modern coordination and communication paradigms and distributed data structures
3. To introduce a variety of methodologies and approaches for reasoning about concurrent and distributed programs.
4. To realize basic principles and best practice engineering techniques of concurrent and distributed computing.
5. To study the safety and progress properties of concurrent and distributed algorithms.
6. To understand the performance of current multi-core and future many-core systems.

Course Outcomes:

After learning the course the student will be able:

1. To identify the core concepts of distributed systems.
2. To learn orchestration of multiple machines to correctly solve problems in an efficient, reliable and scalable way.
3. To examine concepts of distributed systems in designing large systems.
4. To apply distributed computing concepts to develop sample systems.

Course Content:

UNIT I

Introduction: Historical background, Key characteristics, Design goals and challenges, Review of networking and internetworking, Internet protocols.

UNIT II

Processes and Inter process Communication: Processes and threads, Virtualization, Code migration, The API for the Internet protocols, External data representation, Client-server communication, Multicast communication, Message oriented communication, Network virtualization, Overlay networks, RPC and MPI.

UNIT III

Naming: Name services and Domain Name System, Directory services, Case study: X.500 directory service, Time, Global States and Synchronization: Physical and logical clocks, Global states, Mutual exclusion, Election algorithms, Consistency and Replication: Consistency models, Replica management, Consistency protocols, Case studies of highly available services: the gossip architecture and Coda

UNIT IV

Fault Tolerance and Security: Distributed Commit, Recovery, Security Issues, Cryptography. Distributed File Systems: File service architecture, Case study: Sun Network File System, The Andrew File System.

UNIT V

Peer to peer Systems: Introduction, Napster, Peer-to-peer middleware, Routing overlays, Case studies: Pastry, Tapestry. Distributed Object Based Systems: Distributed objects, Java beans, CORBA.

Text Books:

1. Tanenbaum A.S, ***"Distributed Systems: Principles and Paradigms"***, 2nd Edition, Pearson Education, 2006.
2. Coulouris G., Dollimore J., Kindberg T. and Blair G., ***"Distributed Systems: Concepts and Design"***, 5th Edition, Addison Wesley, 2011.
3. Mahajan S., Shah S., ***"Distributed Computing"***, 1st Edition, Oxford University Press, 2010.

Reference Books:

1. Hwang K., Dongarra J., Geoffrey C. Fox, ***"Distributed and Cloud Computing: From Parallel Processing to the Internet of Things"***, Morgan Kaufmann, 2011.
2. Comer D.E. and Droms, R.E., ***"Computer Networks and Internets"***, 4th Edition, Prentice-Hall, 2004.

Course Title:	**GPU Computing**	**Semester VII**	
Course Code	**BTITPE703E**	**Course Type**	**Elective**
Prerequisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Data Science**	**Credits**	**3**

Course Objective:

1. To learn parallel programming with Graphics Processing Units (GPUs).

Course Outcomes:

After learning the course the student will be able:

1. To understand the concepts in parallel programming
2. To implement the programs on GPUs
3. To debug and profile parallel programs.

UNIT I

Introduction: History, Graphics Processors, Graphics Processing Units, GPGPUs. Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel programming, CUDA OpenCL / OpenACC, Hello World Computation Kernels, Launch parameters, Thread hierarchy, Warps / Wavefronts, Thread blocks / Workgroups, Streaming multiprocessors, 1D / 2D / 3D thread mapping, Device properties, Simple Programs.

UNIT II

Memory: Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories.

UNIT III

Synchronization: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU, Functions: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

UNIT IV

Support: Debugging GPU Programs. Profiling, Profile tools, Performance aspects, Streams: synchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based Synchronization - Overlapping data transfer and kernel execution, pitfalls.

UNIT V

Case Studies: Image Processing, Graph algorithms, Simulations, Deep Learning.
Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing

Text book:

1. Brian Tuomanen ***"Hands-On GPU Programming with Python and CUDA"*** 1st edition 2018.

Reference books:

1. David Kirk, Wen-meiHwu; "***Programming Massively Parallel Processors: A Hands-on Approach***"; Morgan Kaufman; 2010 (ISBN: 978-0123814722)
2. Shane Cook; **"*CUDA Programming: A Developer's Guide to Parallel Computing with GPUs"***; Morgan Kaufman; 2012 (ISBN: 978-0124159334)

Course Title:	**Natural Language Processing**	**Semester VII**	
Course Code	**BTITOE704A**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To learn the leading trends and systems in natural language processing.
2. To understand the concepts of morphology, syntax, semantics and pragmatics of the language.
3. To recognize the significance of pragmatics for natural language understanding.
4. To describe simple system based on logic and demonstrate the difference between the semantic presentation and interpretation of that presentation.
5. To describe application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

Course Outcomes:

After learning the course the student will be able:

1. To understand the models, methods and algorithms of statistical Natural Language Processing.
2. To implement probabilistic models in code, estimate parameters for such models and run meaningful experiments to validate such models.
3. To apply core computer science concepts and algorithms, such as dynamic programming.
4. To understand linguistic phenomena and explore the linguistic features relevant to each NLP task.
5. To identify opportunities and conduct research in NLP.
6. To analyze experimental results and write reports.

Course Content:

UNIT I

Introduction to NLP: Introduction to language processing, levels of language analysis, Regular Expressions, Text Normalization, Edit Distance, Regular Expressions, Words, Corpora, Text Normalization, Minimum Edit Distance, N-Grams, Evaluating Language Models, Generalization and Zeros, Smoothing, Kneser-Ney Smoothing, The Web and Stupid Backoff, Advanced: Perplexity's Relation to Entropy, Naive Bayes Classifiers, Training the Naive Bayes Classifier, Worked, Optimizing for Sentiment, Naive Bayes for other text classification, Naive Bayes as a Language Model, Evaluation: Precision, Recall, F-measure, Test sets and Cross-validation, Statistical Significance Testing.

UNIT II

Lexical Semantics, Vector Semantics, Words and Vectors, Cosine for measuring similarity, TF-IDF: Weighing terms in the vector, Applications of the tf-idf vector model, Optional: Pointwise Mutual Information (PMI), Word2vec, Visualizing Embeddings, Semantic properties of embeddings, Bias and Embeddings, Evaluating Vector Models, Feed-Forward Neural Networks, Training Neural Nets, Simple Recurrent Neural Networks, Applications of Recurrent Neural Networks, Deep Networks: Stacked and Bidirectional RNNs, Managing Context in RNNs: LSTMs and GRUs, Words, Subwords and Characters, Encoder-Decoder Networks, Attention, Applications of Encoder-Decoder Networks,

Self-Attention and Transformer Networks, Use of ANN for language modeling and machine translation as case study.

UNIT III

English Word Classes, The Penn Treebank, Part-of-Speech Tagset, Part-of-Speech Tagging, HMM Part-of-Speech Tagging, Maximum Entropy Markov Models, Bidirectionality, Part-of-Speech Tagging for Morphological Rich Languages Constituency Grammars: Constituency, -Free Grammars, Some Grammar Rules for English, Treebanks, Grammar Equivalence and Normal Form, Lexicalized Grammars, Constituency Parsing: Ambiguity, CKY Parsing: A Dynamic Programming Approach, Partial Parsing, Models for Sequential tagging – MaxEnt, CRF, Parsing, PCFG, Dependency parsing.

UNIT IV

Logical Representations of Sentence Meaning: Computational Desiderata for Representations, Model-Theoretic Semantics, First-Order Logic, Event and State Representations, Description Logics, Computational Semantics and Semantic Parsing, Named Entity Recognition, Relation Extraction, Extracting Times, Extracting Events and their Times, Filling, Distributional semantics, Lexical Semantics, Word Senses and WordNet: Word, Relations Between Senses, WordNet: A Database of Lexical Relations, Word Sense Disambiguation, Alternate WSD algorithms and Tasks, Using Thesauruses to Improve Embeddings, Word Sense Induction, Semantic Role Labeling, Semantic Roles, Diathesis Alternations, Semantic Roles: Problems with Thematic Roles, The Proposition Bank, FrameNet, Semantic Role Labeling, Selectional Restrictions, Primitive Decomposition of Predicates.

UNIT V

Lexicons for Sentiment, Affect, and Connotation: Defining Emotion, Available Sentiment and Affect Lexicons, Creating Affect Lexicons by Human Labeling, Semi-supervised Induction of Affect Lexicons, Supervised Learning of Word Sentiment, Using Lexicons for Sentiment Recognition, Other tasks: Personality, Affect Recognition, Lexicon-based methods for Entity-Centric Affect, Connotation Frames. Coreference Resolution: Coreference Phenomena: Linguistic Background, Coreference Tasks and Datasets, Mention Detection, Architectures for Coreference Algorithms, Classifiers using hand-built features, A neural mention-ranking algorithm, Evaluation of Coreference Resolution, Entity Linking, Winograd Schema problems, Gender Bias in Coreference. Discourse Coherence: Coherence Relations, Discourse Structure Parsing, Centering and Entity-Based Coherence, Representation learning models for local coherence, Global Coherence. Applications of NLP: Topic modeling, Query processing, Question answering, Information retrieval and extraction, Entity linking, Text summarization, Conversation modeling, Text entailment.

Text Books:

1. D. Jurafsky& J. H. Martin, ***"Speech and Language Processing – An introduction to Language processing, Computational Linguistics, and Speech Recognition"***, Pearson Education, 3rd Edition, 2019.
2. Allen, James, ***"Natural Language Understanding"***, 2nd Edition, Benjamin/Cummings, 1996.

Reference Books:

1. Bharathi, A., VineetChaitanya and Rajeev Sangal, ***"Natural Language Processing-A Pananian Perspective",*** Prentice Hall India, 1995.
2. Eugene Cherniak, ***"Statistical Language Learning"***, MIT Press, 1993. Manning, Christopher and Heinrich Schütze, ***"Foundations of Statistical Natural Language Processing",*** MIT Press, 1999.

Course Title:	**Artificial Intelligence**	**Semester VII**	
Course Code	**BTITOE704B**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To acquaint the students with the theoretical and computational techniques in Artificial Intelligence.
2. To use various symbolic knowledge representation to specify domains and reasoning tasks of a situated software agent.
3. To use different logical systems for inference over formal domain representations and trace how a particular inference algorithm works on a given problem specification.
4. To understand the conceptual and computational trade-offs between the expressiveness of different formal representations.

Course Outcomes:

After learning the course the students should be able:

1. To find appropriate idealizations for converting real world problems into AI search problems formulated using the appropriate search algorithm.
2. To analyze, formalize and write algorithmic methods for search problem.
3. To explain important search concepts, the definitions of admissible and consistent heuristics and completeness and optimality.
4. To implement and execute by hand alpha-beta search.
5. To design good evaluation functions and strategies for game playing.
6. To carry out proofs in first order and propositional logic using techniques such as resolution, unification, backward and forward chaining.
7. To choose and implement learning algorithms such as decision trees, support vector machines, and boosting.

Course Content:

UNIT I

Introduction: Overview of Artificial intelligence- Problems of AI, AI techniques, Tic - Tac - Toe problem. Intelligent Agents: Agents & environment, Nature of environment, Structure of agents, Goal based agents, Utility based agents, Learning agents.

UNIT II

Problem Solving**:** Problems, Problem Space & search: Defining the problem as state space search, Production system, Problem characteristics and issues in the design of search programs. Search techniques: Solving problems by searching: problem solving agents, Searching for solutions; uniform search strategies: Breadth first search, Depth first search, Depth limited search, Bidirectional search, Comparing uniform search strategies.

UNIT III

Heuristic search strategies: Greedy best-first search, A* search, Memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, Simulated annealing search, Local beam search, Genetic algorithms; Constraint satisfaction problems, Local search for constraint satisfaction problems. Adversarial search: Games, optimal decisions & strategies in games, Theminimax search procedure, Alpha-beta pruning, Additional refinements, Iterative deepening.

UNIT IV

Knowledge & reasoning: Knowledge representation issues, Representation & mapping, Approaches to knowledge representation, Issues in knowledge representation. Using predicate logic: Representing simple fact in logic, Representing instant & ISA relationship, Computable functions & predicates, Resolution, Natural deduction. Representing knowledge using rules: Procedural verses declarative knowledge, Logic programming, Forward verses backward reasoning, Matching, Control knowledge.

UNIT V

Probabilistic reasoning: Representing knowledge in an uncertain domain, The semantics of Bayesian networks, Dempster-Shafer theory, Fuzzy sets & fuzzy logics, Planning: Overview, Components of a planning system, Goal stack planning, Hierarchical planning and other planning techniques, Natural Language processing: Introduction, Syntactic processing, Semantic analysis, Discourse & pragmatic processing. Learning: Forms of learning, Inductive learning, Learning decision trees, explanation based learning, Learning using relevance information, Neural net learning & genetic learning, Expert Systems: Representing and using domain knowledge, Expert system shells and knowledge acquisition.

Text Books:

1. Rich, E. and Knight K., ***"Artificial Intelligence"***, Tata McGraw- Hill.
2. Russell, S. and Norvig P., ***"Artificial Intelligence: A Modern Approach"***, Pearson Education.
3. Patterson, Dan W. ,***"Introduction to Artificial Intelligence & Expert Systems"***, PHI, 2005.

Reference Book:

1. Nilsson, N. J., Morgan Kaufmann, ***"Artificial Intelligence: A New Synthesis"***,Tata McGraw-Hill.

Course Title:	**Advanced Database Technology**	**Semester VII**	
Course Code	**BTITOE704C**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Data Science**	**Credits**	**3**

Course Objectives:

1. To learn the various types of databases and their advanced applications.
2. To understand how and where databases are used in industry.
3. To examine the requirements on special databases.
4. To learn complex queries and interface them with applications.

Course Outcomes:

After learning the course, the students should be able:

1. To explain how databases are used in various fields of industry.
2. To apply query evaluation techniques and query optimization techniques.
3. To develop transaction processing systems with concurrency control.
4. To design and develop a database application system as part of a team.
5. To explore open issues in advanced databases.

Course Content:

UNIT I

Parallel and Distributed Databases: Database System Architectures: Centralized and Client-Server Architectures, Server system Architectures, Parallel Systems, Distributed Systems. Parallel Databases: I/O Parallelism, Inter and Intra Query Parallelism, Inter and Intra operation Parallelism. Distributed Database Concepts, Distributed Data Storage, Distributed Transactions, Commit Protocols, Concurrency Control, Distributed Query Processing, Three Tier Client Server Architecture, Case Studies.

UNIT II

Object and Object Relational Databases: Concepts for object databases: Object identity, Object structure, Type constructors, Encapsulation of operations, Methods, Persistence, Type and Class Hierarchies, Inheritance, Complex Objects, Object Database Standards, Languages and Design: ODMG Model, ODL, OQL, Object Relational and Extended, Relational Systems: Object Relational features in SQL /Oracle, Case Studies.

UNIT III

Xml Databases: XML Data Model, DTD, XML Schema, XML Querying, Web Databases, JDBC, Information Retrieval, Data Warehousing, Data Mining.Mobile Databases: Location and Handoff Management, Effect of Mobility on Data Management, Location Dependent Data Distribution, Mobile Transaction Models, Concurrency Control, Transaction Commit Protocols, Mobile Database Recovery Schemes.

UNIT V

Intelligent Databases: Active databases, Deductive Databases, Knowledge bases, Multimedia Databases, Multidimensional Data Structures, Image Databases, Text/Document Databases, Video Databases, Audio Databases, Multimedia Database Design, Complex Queries and Reasoning: Logic of Query Languages, Relational Calculi, Recursive rules, Syntax and semantics of Datalog, Fix-point semantics, Implementation Rules and Recursion, Rule rewriting methods, Compilation and Optimization, Recursive Queries in SQL, Open issues.

Text Books:

1. Carlo Zaniolo, Stefano Ceri, ***"Advanced Database Systems"***, Morgan Kauffmann Publishers.
2. Subramaniam, ***"Multimedia Databases"***, Morgan Kauffman Publishers, 2008.
3. Rajesh Narang, ***"Object Oriented Interfaces and Databases"***, Prentice-Hall of India, Pvt. Ltd., 2004.
4. Thomas Cannolly and Carolyn Begg, ***"Database Systems, A Practical Approach to Design, Implementation and Management"***, Pearson Education, 3rd Edition, 2007.
5. Jeffrey A. Hoffer, Mary B. Prescottand Fred R. McFadden, ***"Modern Database Management"***, Prentice Hall, 2007.

Reference Books:

1. Henry F Korth, Abraham Silberschatz and S. Sudharshan, ***"Database System Concepts"***, McGraw Hill, 6th Edition, 2011.
2. C. J. Date, A. Kannan and S. Swamynathan, ***"An Introduction to Database Systems"***, Pearson Education, 8th Edition,2006.
3. R. Elmasri, S. B. Navathe, ***"Fundamentals of Database Systems"***, Pearson Education/Addison Wesley, 5th Edition, 2007.
4. Ramakrishnan, Gehrke, ***"Database Management System"***, Tata McGraw Hill Publications, 4th Edition.
5. RamezElmasri, Sham Navathe, ***"Fundamentals of Database Systems"***, Addison-Wesley, 2000.

Course Title:	**Machine Learning Lab**	**Semester VII**	
Course Code	**BTITL705**	**Course Type**	**Compulsory**
Prerequisite	**NIL**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Study and Implement the Naive Bayes learner using WEKA. (The datasets taken can be: Breast Cancer data file).
2. Study and Implement the Decision Tree learners using WEKA. (The datasets taken can be: Breast Cancer data file or Reuter's data set).
3. Estimate the accuracy of decision classifier on breast cancer dataset using 5-fold cross-validation. (You need to choose the appropriate options for missing values).
4. Estimate the precision, recall, accuracy, and F-measure of the decision tree classifier on the text classification task for each of the 10 categories using 10-fold cross-validation.
5. Develop a machine learning method to classifying your incoming mail.
6. Develop a machine learning method to Predict stock prices based on past price variation.
7. Develop a machine learning method to predict how people would rate movies, books, etc.
8. Develop a machine learning method to Cluster gene expression data, how to modify existing methods to solve the problem better
9. Select two datasets. Each dataset should contain examples from multiple classes. For training purposes assume that the class label of each example is unknown (if it is known, ignore it). Implement the K-means algorithm and apply it to the data you selected. Evaluate performance by measuring the sum of Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameter k.

Course Title:	**Pattern Recognition Lab**	**Semester VII**	
Course Code	**BTITL706**	**Course Type**	**Elective**
Prerequisite	**-**	**L – T – P**	**0– 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
2. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.
3. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
4. Write a program to construct aBayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
5. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
6. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
7. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
8. Study of Feature Selection techniques

Course Title:	**Soft Computing Lab**	**Semester IV**	
Course Code	**BTITL706**	**Course Type**	**Elective**
Prerequisite	**-**	**L – T – P**	**0 – 0 – 2**
Stream	**Laboratory Course**	**Credit**	**1**

Lab Experiments List:

1. Create a perceptron with appropriate no. of inputs and outputs. Train it using fixed increment learning algorithm until no change in weights is required. Output the final weights.
2. Create a simple ADALINE network with appropriate no. of input and output nodes. Train it using delta learning rule until no change in weights is required. Output the final weights.
3. Train the autocorrelator by given patterns: A1=(-1,1,-1,1), A2=(1,1,1,-1), A3=(-1, -1, - 1, 1). Test it using patterns: Ax=(-1,1,-1,1), Ay=(1,1,1,1), Az=(-1,-1,-1,-1).
4. Train the hetrocorrelator using multiple training encoding strategy for given patterns: A1= (000111001) B1= (010000111), A2=(111001110) B2=(100000001), A3=(110110101) B3(101001010). Test it using pattern A2.
5. Implement Union, Intersection, Complement and Difference operations on fuzzy sets. Also create fuzzy relation by Cartesian product of any two fuzzy sets and perform max-min composition on any two fuzzy relations.
6. Solve Greg Viot's fuzzy cruise controller using MATLAB Fuzzy logic toolbox.
7. Solve Air Conditioner Controller using MATLAB Fuzzy logic toolbox
8. Implement TSP using GA.

Course Title:	**Artificial Neural Networks Lab**	**Semester VII**	
Course Code	**BTITL706**	**Course Type**	**Elective**
Prerequisite	**-**	**L – T – P**	**0– 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Study of Biological Neural Network & Artificial Neural Network.
2. Implementation of logic gate (AND, OR, NOT, NAND, NOR) using McCulloh- Pitts model.
3. Implementation of Perceptron model
4. To design and train a perceptron for identifying ODD and EVEN number
5. Implementation of Back propagation learning algorithm.
6. Write a MATLAB program to show Back Propagation Network for XOR function with Binary Input and Output.
7. Implementation of multilayer perceptron (XOR gate)
8. Study of pattern classification & pattern clustering
9. To create a Bi-directional Associative Memory (BAM) for ID and telephone number.
10. To design and train the Hopfield net to map the input vector with the stored vector and correct them.
11. Write a MATLAB program to illustrate ART neural network.

Course Title:	**Real Time System Lab**	**Semester VII**	
Course Code	**BTITL706**	**Course Type**	**Elective**
Prerequisite	**Operating System**	**L – T – P**	**0– 0 – 2**
Stream	**Core**	**Credit**	**1**

Lab Experiments List:

1. Study of RTOS
2. Write the pseudo code in Linux / Unix using c/c++ to perform a) Round Robin b) SJF c) FCFS d) Priority
3. Implement all file allocation strategies
4. Implementation of IPC using Pipe Processing
5. Design a Real Time Implementation of Real Time Video Surveillances System.
6. Design and Implementation to operate Calculator using Touch Screen
7. Design a Real time Clock
8. Design and Develop the code for controlling traffic light at an intersection.
   Consider an intersection with two, two-way streets. A traffic light will normally be green G second, yellow for Y seconds and red R seconds During the light for certain period of time, the intersection will automatically suspend normal service and its light will flash yellow.
   a) Develop an object-oriented design.
   b) Using programming language
   c) Make suitable assumption and state them clearly.
9. Interface a Seven Segment Display.

Course Title:	**Information Security Lab**	**Semester**	**VII**
Course Code	**BTITL706**	**Course Type**	**Elective**
Pre-requisite	**Computer Networks and internetworking Lab**	**L – T – P**	**0 – 0 – 2**
Stream		**Credit**	**1**

Lab Experiments List:

1. Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts:
    1. Caesar Cipher
    2. Playfair Cipher
    3. Hill Cipher
    4. Vigenere Cipher
    5. Rail fence- Row & Column Transformation.
2. Implement the following algorithms:
    1. DES
    2. RSA Algorithm
    3. Diffie-Hellman
    4. MD5
    5. SHA-1
3. Implement the SIGNATURE SCHEME - Digital Signature Standard.
4. Demonstrate how to provide secure data storage, secure data transmission for creating digital signatures (GnuPG).
5. Setup a honeypot and monitor the honeypot on network (KFSensor).
6. Installation of rootkit and study about the variety of options.
7. Perform wireless audit on an access point or a router and decrypt WEP and WPA (Net Stumbler).
8. Demonstrate Intrusion Detection System (IDS) using any tool (snort or any other s/w).

Course Title:	**Electronic Payment Systems Lab**	**Semester**	**VII**
Course Code	**BTITL706**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**NIl**	**Credit**	**1**

Assignment & Projects based on syllabus.

Course Title:	**Distributed Computing Lab**	**Semester**	**VII**
Course Code	**BTITL706**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Nil**	**Credit**	**1**

Lab Experiments List:

1. Case study on Common Object Request Broker Architecture.
2. Implementation of Deadlock through Simulation.
3. Study of 3 tier client server architecture.
4. Case study on Client and RMI Server.
5. WAP to Implement an Election algorithm.
6. S/W Simulation for Clock Synchronization in Distributed System using Lamport's Algorithm.
7. Implementation of Banker's Algorithm for avoiding Deadlock
8. Simulate the following CPU scheduling algorithms a) Round Robin b) SJF c) FCFS d) Priority
9. Simulate all page replacement algorithms a) FIFO b) LRU c) LFU
10. Implement shared memory and semaphore concepts for inter process communication

Course Title:	**GPU Computing Lab**	**Semester**	**VII**
Course Code	**BTITL706**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Nil**	**Credit**	**1**

Lab Experiments List:

1. Program to perform Vector addition
2. Program to perform Matrix-Matrix Multiplication
3. Program to perform Numberical calculation of value of pi
4. Program to perform Vector Reduction
5. Program to perform Matrix-Matrix Multiplication with Tiling and Shared Memory
6. Program to perform Vector Reduction with Unlimited Input Elements
7. Program to perform Matrix-Matrix Multiplication with Performance Tuning
8. Program to perform MRI with Performance Tuning

Course Title:	**Database Programming**	**Semester VII**	
Course Code	**BTITAO01**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Course Objective:

1. Introduction to Relational Database Management
2. To learn what is Entity-Relationship Diagrams and significance of it
3. To learn Normalization and its usages in real world. Impact of denormalization.
4. To learn SQL for Data Retrieval and Manipulation and explore its usages to meet the business need
5. To learn how to manage data integrity using Constraints
6. Query Optimization using indexes
7. Develop efficient PL/SQL programs for data manipulation
8. Design Modular application using packages
9. Invoke Native dynamic SQL to build runtime SQL statements
10. Learn advance programming techniques such as cursors and cursor variables
11. Enhance performance using collection datatypes and bulk operations
12. Create database triggers to solve business challenges and to enforce business rules
13. To manage various database objects and dependencies

Course Outcome:

After completing this course students would be able:

1. To understand the importance of Database Management System in real world
2. To design Database and required components as per business need by using the knowledge of Normalizations, Entity-Relationship, Constraints, Procedures, Functions, Packages, Triggers, Views etc
3. To manipulate data using SQL (DML/DDL) statements, procedures, functions, packages, views, triggers etc
4. To understand performance issues and its resolution.

Course Content:

UNIT I

Overview of Database Management system and Programming., Database models: Flat, Hierarchical, Network, Relational., Entity-Relationship Diagrams.

Relational database design. Types of database design.

Relationships: One-to-One, One-to-Many, Many-to-Many.

Normalization: First, Second, Third and Higher Normal Forms, Denormalization.

Integrity rules. What is Data, Data Definition: Schemas and User, Data Dictionary.

Schemas, Types of Schema. Database Users.

User Management- Database User creation, Privileges, Roles and Roles Management.

UNIT II

DDL statements for Table, Object Types, Users and Schema, Object Namespaces

Understanding table design. List of various datatypes available for Columns of the table. Create Table, Alter Table, Dropping and Truncating tables.

Constraints, Types of Constraints, Constraint Creation. Addition of Constraints. View Constraints Information, Constraint Management (Enable/Disable, Dropping).

Data Retrieval using Select statement. Usages of Arithmetic Operators. SQL Expressions and Operators.

Data Restricting and Sorting: Limit the rows, Comparison Operators. LIKE operator, Boolean Operator, NULL values, Precedence Rule. Data sorting. Substitution variables.

Single Row Functions: Defining a Function, Types of Function, Use of Character, Number and Date Functions.

Conversion Functions and Conditional Expressions. TO_CHAR, TO_DATE, TO_NUMBER. Apply Conditional Expressions like NULLFIF, NVL2, COALESCE, DECODE, CASE expression.

UNIT III

Reporting Aggregated Data using Group Functions. Describe and Identify Group Functions. Data Grouping Restricting using GROUP BY and HAVING clause.

Joins: Equality Join, Non-Equality Join, Outer Joins, Cartesian Join, Self-Join.

Set operators and types, Sets and Venn Diagrams.

Define Subqueries. Types of Subqueries: Single -row, Multiple-row, Co-related and Nested subqueries.

Data Manipulation Overview and Commands (DML). INSERT, UPDATE, DELETE, MERGE, TRUNCATE. Insert, update and delete rows from the table.

What is Transaction. Transaction Controlling using COMMIT, ROLLBACK and SAVEPOINT. Lock Table, SELECT …FOR UPDATE command.

View Introduction, Simple and Complex view, View creation. Data Manipulation using View, Dropping View, Materialized View.

Other database objects like Sequences, Indexes and Synonyms.

UNIT IV

PL/SQL Programming Language Overview. PL/SQL processing Architecture, Two-Tire, N-Tire Model, PL/SQL Block Structure, Types of PL/SQL blocks.

Delimiters, Identifiers, Literals, Comments, Variable and Data types

Control Structure: Conditional Statements (IF statements, CASE statements, GOTO and NULL statements),

Iterative Processing Statements: Simple Loop, WHILE Loop, FOR Loop,

Cursor Structures: Implicit Cursors, Explicit Cursors, Cursor Attributes, Cursor FOR LOOP, Parametric Cursor, FOR UPDATE and WHERE CURRENT Clauses.

Bulk Statements: BULK COLLECT INTO, FOR ALL

Dynamic SQL: Architecture, Native Dynamic SQL, DBMS_SQL package

Object Types: Declare, Implement, White Listing, Getter and Setter, Methods, Inheritance and Polymorphism, Implement Object Type collection.

Triggers: Trigger Architecture, DDL , DML Triggers, Instead of Triggers, System and Database

Event Triggers. Trigger Restrictions: Trigger size, SQL statements, Long Data Types, Mutating Tables. Managing Triggers (Enable, Disable, Removal).

UNIT V

Collections Introductions. Object Types: Varray Collection, Table Collections, Collection Methods

Error Management - Exception Types, Compilation Error, Runtime Error, Exception Management using Built-in Functions, User Defined Exceptions.

Functions and Procedures. Calling Subroutines, Positional Notation, Named Notation, Mixed Notation, SQL Call Notation. Function and Procedure Creation, Pass-by-Value, Pass-by-Reference.

Packages – Package Architecture. Package Specification: Variables, Types, Components (Function and Procedure), Serially Reusable Precompiler Directive. Package Body: Variables, Types, Components (Procedure and Function). Definer vs. Invoker Rights, Managing Packages in Database: Finding, Validating and Describing, Checking Dependencies, Overloading subroutines.

Transaction Management: COMMIT, ROLLBACK, SAVEPOINT. LOCKING, Transaction Isolation, Deadlocks.

Large Objects: Introduction, LOB Types, SQL to work with LOBs (CLOB, NCLOB, BLOB, BFILE), Use of PL/SQL to work with LOBs.

SQL Tuning: EXPLAIN PLAN statement, DBMS_XPLAN package.

SQL Tracing: Tracing Session Statement, Convert Raw Trace File to Readable format

Text Books:

1. OCA Oracle Database 12c SQL Fundamentals I Exam Guide (Exam 1Z0-061), 2nd Edition by RoopeshRamklass, Published by Oracle Press, 2014
2. Oracle database 12c PL/SQL Programming by Michael McLaughlin, McGraw Hill Education, 2014.

Reference Books:

1. J. Murach, “Murach’s Oracle SQL and PL/SQL: Works with All Versions Through 11g”, Shroff Publication, 2008.
2. Oracle Database SQL Language Reference, 12c Release 2 (12.2), part of Oracle Documentation.
3. Oracle Database PL/SQL Language Reference, 12c Release 2 (12.2), part of Oracle Documentation.

Course Title: Deep Neural Networks **Semester VII**

Course Code	**BTITAO02**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus prepation is in progress.

Course Title:	**Compliance, risk and Governance: IT Perspective**	**Semester VII**	
Course Code	**BTITAO03**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Course Objective:

1. To know the goals and objectives of IT GRC.
2. Deeper dive into concepts, tools and applications for each of areas within GRC.
3. Introduction to key processes, tools, methodologies, and framework – as applicable for GRC.
4. To learn the fundamentals of SOX.

Course Outcome:

After completing this course students would be able:

1. To describe the need for IT GRC.
2. To be aware of the basic building blocks of GRC and apply the learned knowledge as well relate to real scenarios in IT industry with respect to GRC.
3. To explain the mandatory items that need to be checked.
4. Familiarize themselves with actual SOX controls, TOD and TOE as used in IT industry.

UNIT I

IT Governance: Introduction, fundamentals, why and how. Role of IT Governance.Peek into COBIT, ITIL, COSO, CMMI, FAIR.

UNIT II

IT Risk: Risk Response, Risk Identification, Assessment and Evaluation, Risk Response, Risk Monitoring, Information Systems Control Design and Implementation.

UNIT III

Information Systems Control Monitoring and Maintenance.

UNIT IV

IT Compliance: Introduction, Internal and External Compliance, Regulatory Compliance Standards.

UNIT V

IT Compliance Goals and Challenges, Peek into SOX.GLBA, FISMA, HIPAA, PCI DSS, SSAE 16, Basel III, The Sarbanes -Oxley Act (SOX) : Deeper dive in context of IT, Control areas, List of actual controls, applicability, Test of Design (TOD), Test of Execution (TOE).

Text Books:

1. To be confirmed.

Reference Books:

1. To be confirmed.

Course Title:	**ERP: Oracle Application**	**Semester VII**	
Course Code	**BTITAO04**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Database Programming Lab**	**Semester VII**	
Course Code	**BTITAOL01**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Deep Neural Network Lab**	**Semester VII**	
Course Code	**BTITAOL02**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**Compliance, risk and Governance: IT Perspective Lab**	**Semester VII**	
Course Code	**BTITAOL03**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Course Title:	**ERP: Oracle Application Lab**	**Semester VII**	
Course Code	**BTITAOL04**	**Course Type**	**Add-on**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream		**Credits**	**3**

Syllabus preparation is in progress.

Honors Courses (For B.Tech. Programme in IT; any Five Courses)

Course Title:	**Internet of Things**	**Semester**	
Course Code	**BTITH01**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors Courses**	**Credits**	**3**

Course Objectives:

1. To understand the vision of IoT.
2. To understand IoT market perspective.
3. To study the data and knowledge management and use of devices in IoT technology.
4. To understand state of the art – IoT Architecture.
5. To study the real world IoT design constraints, industrial automation and commercial building automation in IoT.

Course Outcomes:

After learning the course the students should be able:

1. To interpret the vision of IoT from a global context.
2. To determine the market perspective of IoT.
3. To compare and contrast the use of devices, gateways and data management in IoT.
4. To implement state of the art architecture in IoT.
5. To illustrate the application of IoT in industrial automation and identify real world design constraints.

Course Content:

UNIT I

M2M to IoT: The Vision-Introduction, From M2M to IoT, M2M towards IoT: the global context, A use case example,Differing characteristics.

UNIT II

M2M to IoT: A Market Perspective– Introduction, Definitions, M2M Value Chains, IoT Value Chains, An emerging industrialstructure for IoT, The international driven global value chain and global information monopolies, M2M to IoT. Anarchitectural overview: Building an architecture, Main design principles and needed capabilities, An IoT architecture outline,Standards considerations.

UNIT III

M2M and IoT Technology Fundamentals: Devices and gateways, Local and wide area networking, Data management,Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management.

UNIT IV

IoT Architecture: State of the Art, Introduction, State of the art, Architecture Reference Model - Introduction, Referencemodel and architecture, IoT reference model, Industrial Automation: Service-oriented architecture-based device integration, SOCRADES: realizing the enterpriseintegrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, Commercial Building Automation

UNIT V

IoT Reference Architecture: Introduction, Functional view, Information view, Deployment and operational View, Otherrelevant architectural views. Real-World Design Constraints - Introduction, Technical design constraints-hardware is popularagain, Data representation and visualization, Interaction and remote control.

Introduction Case study: phase one-commercial building automation today, Case study: phase two-commercial building automation in the future.

Text Book:

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle,"***From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence***", Academic Press, 1st Edition, 2014.

Reference Books:

1. Vijay Madisetti, Arshdeep Bahga, "***Internet of Things (A Hands-on-Approach)",*** VPT, 1st Edition, 2014.
2. Francis da Costa, "***Rethinking the Internet of Things: A Scalable Approach to Connecting Everything***", 1st Edition, Apress Publications, 2013.

Course Title: Big Data Computing		**Semester**	
Course Code: BTITH02		**Course Type**	**Honors**
Pre-requisite: Nil		**L – T – P**	**3 – 0 – 0**
Stream:	**Honors courses**	**Credits**	**3**

Course Objectives:

1. To understand the concept of Big Data.
2. To learn Big Data file systems and their storage methods.
3. To learn to process Big Data information for analytics.
4. To discuss and understand Big Data implementations within large corporations like Google and Facebook.

Course Outcomes:

After learning the course the students should be able:

1. To model and implement efficient big data solutions for various application areas using appropriately selected algorithms and data structures.
2. To analyze methods and algorithms to compare and evaluate them with respect to time and space requirements and make appropriate design choices when solving real-world problems.
3. To explain trade-offs in big data processing techniques.
4. To explain the Big Data fundamentals including the evolution of Big Data, the characteristics of Big Data and the challenges introduced.
5. To apply non-relational databases techniques for storing and processing large volumes of structured and unstructured data, as well as streaming data.
6. To apply the novel architectures and platforms introduced for Big data in particular Hadoop and MapReduce.

Course Content:

UNIT I

Introduction to Big Data: Why Big Data and Where did it come from?, Characteristics of Big, DataVolume, Variety, Velocity, Veracity, Valence, Value, Challenges and applications of Big Data.

UNIT II

Introduction to Enabling Technologies for Big Data, Introduction to Big Data Stack, Introduction to some Big Data distribution packages.

UNIT III

Introduction to Big Data Platforms, Overview of Apache Spark, HDFS, YARN, Introduction to MapReduce, MapReduce Programming Model with Spark, MapReduce Example: Word Count, Page Rank, Big data Machine learning with Spark, Big Data Machine Learning Algorithms in Spark, Introduction to Spark MLlib.

UNIT IV

Introduction to Big Data Storage Platforms for Large Scale Data Storage, CAP Theorem, Eventual Consistency, Consistency Trade-O-s, ACID and BASE, Introduction to Zookeeper and Paxos,Introduction to Cassandra, Cassandra Internals, Introduction to HBase, HBase Internals, Introduction to Pregel, Introduction to Giraph, Introduction to Spark GraphX.

UNIT V

Introduction to Big Data Streaming Platforms for Fast Data, Introduction to Big Data Streaming Systems, Big Data Pipelines for Real-Time computing, Introduction to Spark Streaming, Kafka, Streaming Ecosystem. Introduction to Big Data Applications (Machine Learning),Overview of Big Data Machine Learning, Mahout Introduction, Big Data Machine learning Algorithms in Mahout-kmeans, Naïve Bayes, Introduction to Deep Learning for Big Data. Introduction to Big Data Applications (Graph Processing).

Text Book:

1. Anand Rajaraman and Jeffrey Ullman, ***"Mining of Massive Datasets"***, Cambridge University Press, 2012.

Reference Books:

1. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, ***"An Introduction to Information Retrieval"***, Cambridge University Press, 2008.
2. Jimmy Lin and Chris Dyer, ***"Data-Intensive Text Processing with MapReduce"***, Morgan and ClayPool Publishers, 2010.

Course Title:	**Block Chain Technology**	**Semester**	
Course Code	**BTITH03**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors Courses**	**Credits**	**3**

Course Objectives

1. Understand how blockchain systems (mainly Bitcoin and Ethereum) work,
2. To securely interact with them,
3. Design, build, and deploy smart contracts and distributed applications,
4. Integrate ideas from blockchain technology into their own projects.

Course Outcomes

1. Explain design principles of Bitcoin and Ethereum.
2. Explain Nakamoto consensus.
3. Explain the Simplified Payment Verification protocol.
4. List and describe differences between proof-of-work and proof-of-stake consensus.
5. Interact with a blockchain system by sending and reading transactions.
6. Design, build, and deploy a distributed application.
7. Evaluate security, privacy, and efficiency of a given blockchain system.

Course Content:

UNIT I

Basics: Distributed Database, Two General Problem, Byzantine General problem and Fault Tolerance, Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing Complete. Cryptography: Hash function, Digital Signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof.

UNIT II

Blockchain: Introduction, Advantage over conventional distributed database, Blockchain Network, Mining Mechanism, Distributed Consensus, Merkle Patricia Tree, Gas Limit, Transactions and Fee, Anonymity, Reward, Chain Policy, Life of Blockchain application, Soft & Hard Fork, Private and Public blockchain, Naive Blockchain construction: Memory Hard algorithm – Hashcash implementation, Direct Acyclic Graph, Play with Go-ethereum, Smart Contract Construction, Toy application using Blockchain, Mining puzzles.

UNIT III

Distributed Consensus: Nakamoto consensus, Proof of Work, Proof of Stake, Proof of Burn, Difficulty Level, Sybil Attack, Energy utilization and alternate.

UNIT IV

Cryptocurrency: History, Distributed Ledger, Bitcoin protocols - Mining strategy and rewards, Ethereum - Construction, DAO, Smart Contract, GHOST, Vulnerability, Attacks, Sidechain, Namecoin.

UNIT V

Cryptocurrency Regulation: Stakeholders, Roots of Bit coin, Legal Aspects-Crypto currency Exchange, Black Market and Global Economy. Applications: Internet of Things, Medical Record Management System, Domain Name Service and future of Blockchain.

Text Book:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, ***"Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction"*** ,Princeton University Press (July 19, 2016).

Reference Books

1. Antonopoulos, **"*Mastering Bitcoin: Unlocking Digital Cryptocurrencies"***
2. Satoshi Nakamoto, **"*Bitcoin: A Peer-to-Peer Electronic Cash System"***
3. DR. Gavin Wood, ***"ETHEREUM: A Secure Decentralized Transaction Ledger"***,Yellow paper.2014.
4. Nicola Atzei, Massimo Bartoletti, and Tiziana Cimoli, **"*A survey of attacks on Ethereum smart contracts"***

Course Title:	**Data Science**	**Semester VIII**	
Course Code	**BTITH04**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors Courses**	**Credits**	**3**

Course Objectives

1. Apply quantitative modeling and data analysis techniques to the solution of real world business problems, communicate findings, and effectively present results using data visualization techniques.
2. Recognize and analyze ethical issues in business related to intellectual property, data security, integrity, and privacy.
3. Apply ethical practices in everyday business activities and make well-reasoned ethical business and data management decisions.
4. Demonstrate knowledge of statistical data analysis techniques utilized in business decision making.

Course Outcomes

1. Develop in depth understanding of the key technologies in data science and business analytics: data mining, machine learning, visualization techniques, predictive modeling, and statistics.
2. Practice problem analysis and decision-making.
3. Gain practical, hands-on experience with statistics programming languages and big data tools through coursework and applied research experiences.

UNIT I

Introduction to Data Mining : Introduction to Data mining and knowledge discovery, Relation to Statistics, Databases, Data Mining Functionalities, Steps In Data Mining Process, Architecture of a Typical Data Mining Systems, Classification of Data Mining Systems.

UNIT II

Vectors and Matrices: Vectors. Create, name and select elements from vectors. Learn how to work with matrices, basic computations with them and demonstrate your knowledge by analyzing the Star Wars box office figures.

UNIT III

Factors & Data Frames: Storing Categorical data in factors. Learn how to create, subset and compare categorical data.

UNIT VI

Data Analysis: Simple regression, Multiple Regression, Multivariate Regression Analysis, Robust Regression, Correlation, Clustering**.**

UNIT V

Data Visualization: R graphics, Plotting, Scatter Plots Bar Charts and Plots 3D graphics**.**
Machine Learning: Data Partitioning Predicting events with machine learning, Supervised and Unsupervised learning.

Text Books:

1. Rajendra Patil, Hiren dand, Rupali Dahake, "***A practical approach to R Tool"***, SPD Publication, 1st Edition, 2017

Reference Books:

1. Joel Grus,**'** ***Data Science from Scratch: First Principles with Python"***, O'Reilly Media, 1st Edition, 2015.
2. Hadley Wickham, Garrett Grolemund, '***R for Data Science Import, Tidy, Transform, Visualize, and Model Data"***, O'Reilly Media, 1st Edition, 2017.
3. Nina Zumel, John Mount, "**Practical Data Science with R**" , Manning, 2014.

Course Title:	**Computer Vision**	**Semester**	
Course Code	**BTITH05**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors Courses**	**Credits**	**3**

Course Objectives:

1. To learn Computer Vision and its applications

Course Outcomes:

After learning the course the student will be able:

1. To compare human and computer vision system.
2. To enhance the image quality by applying different transforms.
3. To segment the image to identify the region of interest.
4. To develop an algorithm to recognize the specified objects in the given image.
5. To classify the data as per the labeled classes.
6. To Apply optical flow motion estimation technique to estimate the object motion in videos.

UNIT I

Image Formation and Low-Level Processing: Human Vision System, Computer Vision System, Fundamentals of Image Formation, Stereo Vision.

UNIT II

Transformation: Orthogonal, Euclidean, Affine, Projective, Convolution and Filtering, Image Enhancement, Histogram Processing.

UNIT III

Feature Extraction: Edges - Canny, LOG, DOG, Line detectors (Hough Transform), Harris Corner detector, SIFT, HOG, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Feature Matching and tracking.

UNIT IV

Object Recognition: Global Methods, Active Contours; Split and Merge, Mean Shift and Mode Finding; Normalized Cuts, Histogram of Oriented Gradients, Change Detection, Motion Estimation: Triangulation, Two-frame structure from motion, Optical flow, Tracking.

UNIT V

Clustering: K-Means, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised Classifiers, Bayes, KNN, ANN models, Dimensionality Reduction: PCA, LDA, ICA.

Text books:

1. Milan Sonka, Vaclav Hlavac, Roger Boyle ***"Image Processing, Analysis, and Machine Vision***,"

Thomson Learning
2. Robert Haralick and Linda Shapiro ***"Computer and Robot Vision"***, , Vol I, II, Addison-Wesley, 1993.
3. Dana H Ballard and Christopher M. Brown, ***"Computer Vision***", PrenticeHall.

Reference Books:

1. Richard Szeliski, ***"Computer Vision: Algorithms and Applications"***, Springer
2. Forsyth, Ponce, ***"Computer Vision-A Modern Approach"***, Low Price Edition, Pearson Education.
3. Bernd Jahne and Host HauBecker, Elsevier. ***"Computer Vision and applications-A Guide for Students and Practitioners"***

Course Title:	**Advanced Machine Learning**	**Semester**	
Course Code	**BTITH06**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors courses**	**Credits**	**3**

Course Objectives:

1. To introduce key concepts in pattern recognition and machine learning; including specific algorithms for classification, regression, clustering and probabilistic modeling.
2. To give a broad view of the general issues arising in the application of algorithms to analysing data, common terms used, and common errors made if applied incorrectly.
3. To demonstrate a toolbox of techniques that can be immediately applied to real world problems, or used as a basis for future research into the topic.

Course Outcomes:

After learning the course the student will be able:

1. To understand the Key concepts, tools and approaches for pattern recognition on complex data sets
2. To understand the Kernel methods for handling high dimensional and non-linear patterns.
3. To understand the State-of-the-art algorithms such as Support Vector Machines and Bayesian network.
4. To Solve real-world machine learning tasks: from data to inference.
5. To understand the Theoretical concepts and the motivations behind different learning frameworks.

UNIT I

Key concepts, Supervised/Unsupervised Learning, Recent trends in supervised and unsupervised learning algorithm, dimensional reducibility, feature selection and extraction**.**

Loss functions and generalization, Probability Theory, Parametric vs Non-parametric methods, Elements of Computational Learning Theory Ensemble Learning, Bagging, Boosting, Random Forest.

UNIT II

Kernel Methods for non-linear data, Support Vector Machines, Kernel Ridge Regression, Structure Kernels, Kernel PCA, Latent Semantic Analysis.

UNIT III

Bayesian methods for using prior knowledge and data, Bayesian inference, Bayesian Belief Networks and Graphical models, Probabilistic Latent Semantic Analysis, The Expectation-Maximisation (EM) algorithm, Gaussian Processes.

UNIT IV

Dimensionality Reduction - CCA, LDA, ICA, NMF – Canonical Variates - Feature Selection vs Feature Extraction.

UNIT V

Filter Methods - Sub-space approaches - Embedded methods Low-Rank approaches - Recommender Systems.Application areas - Security - Business – Scientific.

Text books:

1. Christopher M. Bishop, Pattern Recognition and Machine Learning.
2. John Shawe-Taylor and NelloCristianini, Kernel Methods for Pattern Analysis.

Course Title:	**Recommender Systems**	**Semester**	
Course Code	**BTITH07**	**Course Type**	**Honors**
Prerequisite	**Nil**	**L – T – P**	**3-0-0**
Stream	**Honors Cources**	**Credits**	**3**

Course Objectives:

1. To provide in depth knowledge of Recommender Systems
2. To provide knowledge of Context-Aware Recommender Systems
3. To find an in-depth knowledge Content-based Recommender Systems: State of the Art and Trends

Course Outcomes:

After learning the course the students should be able:

1. To develop an ability for decision-making processes, such as what items to buy, what music to listen, or what news to read
2. To help online users to cope with the information overload and have become one of the most powerful and popular tools in electronic commerce.

Course Content:

UNIT I

Introduction to Recommender Systems: Introduction, Recommender Systems Function, Data and Knowledge Sources, Recommendation Techniques, Application and Evaluation, Recommender Systems and Human Computer Interaction: Trust, Explanations and Persuasiveness, Conversational Systems, Visualization; Recommender Systems as a Multi-Disciplinary Field, Emerging Topics and Challenges.

UNIT II

Data Mining Methods for Recommender Systems: Data Preprocessing: Similarity Measures, Sampling, Reducing Dimensionality, Denoising; Classification: Nearest Neighbors, Decision Trees, Ruled-based Classifiers, Bayesian Classifiers, Artificial Neural Networks, Support Vector Machines, Ensembles of Classifiers, Evaluating Classifiers; Cluster Analysis: k-Means, Alternatives to k-means; Association Rule Mining.

UNIT III

Content-based Recommender Systems: State of the Art and Trends: Basics of Content-based Recommender Systems: A High Level Architecture of Content-based Systems, Advantages and Drawbacks of Content-based Filtering; State of the Art of Content-based Recommender Systems: Item Representation, Methods for Learning User Profiles; Trends and Future Research: The Role of User Generated Content in the Recommendation Process, Beyond Over-specialization: Serendipity.

UNIT IV

Advances in Collaborative Filtering: Preliminaries: Baseline predictors, The Netflix data, Implicit feedback; Matrix factorization models: SVD, SVD++, Time-aware factor model; Neighborhood models: Similarity measures, Similarity-based interpolation, Jointly derived interpolation weights; Enriching neighborhood models: A global neighborhood model, A factorized neighborhood model, Temporal dynamics at neighborhood models.

UNIT V

Context-Aware Recommender Systems: Context in Recommender Systems: What is Context?, Modeling Contextual Information in Recommender Systems, Obtaining Contextual Information; Paradigms for Incorporating Context in Recommender Systems: Contextual Pre-Filtering, Contextual Post-Filtering, Contextual Modeling; Combining Multiple Approaches: Case Study of Combining Multiple Pre-Filters: Algorithms, Case Study of Combining Multiple Pre-Filters: Experimental Results, Additional Issues in Context-Aware Recommender Systems .

Text Book:

1. Francesco Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor, **"*Recommender Systems Handbook*"**, Springer.

Reference Books:

1. Charu C. Aggarwal, **"*Recommender system*"**, Springer.
2. Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich ***"Recommender system"*,** Cambridge University**.**

Course Title:	**Reinforcement Learning**	**Semester**	
Course Code	**BTITH08**	**Course Type**	**Honors**
Pre-requisite	**Machine Learning**	**L – T – P**	**3 – 0 - 0**
Stream	**Honours Course**	**Credits**	**3**

Course Objectives:

1. To understand framework for modeling an autonomous agent's interaction with an unknown world.
2. To learn the effects of agent's actions and modify its policy in order to maximize future reward.

Course Outcomes:

After learning the course the students should be able to:

1. Understand some of the foundational ideas on which modern reinforcement learning is built, including Markov decision processes, Monte Carlo estimation, temporal difference learning.
2. Develop an understanding of the concepts while focusing on the mathematical theory of reinforcement learning.

Course Content:

UNIT I

Overview of reinforcement learning: Agent environment framework, Successes of reinforcement learning, Bandit problems and online learning.

UNIT II

Markov decision processes, Returns and value functions.

UNIT III

Solution methods: dynamic programming, Monte Carlo learning.

UNIT IV

Solution methods: Temporal difference learning, Eligibility traces.

UNIT V

Value function approximation (function approximation), Models and planning (table lookup case).
Case studies: Successful examples of RL systems, Frontiers of RL research.

Text Book:

1. Richard S. Sutton and Andrew G. Barto, ***"Reinforcement Learning - An Introduction (Adaptive Computation and Machine Learning series)"***, MIT Press, 2nd edition, 2018.

Reference Book:

1. Csaba Szepesvari, ***"Algorithms for Reinforcement learning"***, Morgan and Claypool publishers, 2010.

Minor Courses (For other B.Tech. Programme; any Five Courses)

1. Computer Architecture and Organization
2. Data Structures and Applications
3. Database Management Systems
4. Operating Systems
5. Software Engineering
6. Machine Learning
7. Computer Networks and Internetworking Protocols
8. Web Technology