Indian Institute of Technology Indore

Curriculum and Courses of Study for Bachelor of Technology : Computer Science and Engineering

November 2025
[After incorporating decisions of the 58th meeting of the Senate held on 26 November 2025]

CONTENTS

Curriculum of 1st Year BTech (For AY 2009-10)

Semester I

Course Code	Course Title	Weekly Contact Hours (L-T-P)	Credits
CH 101	Chemistry	2-1-0	6
CS 101	Computer Programming & Utilization	2-0-2	6
MA 101	Calculus	3-1-0	8
PH 101	Physics –I	2-1-0	6
CH 151	Chemistry Lab	0-0-3	3
ME 151	Engineering Graphics & Drawing	0-1-3	5
NC 101#	National Cadet Crops (NCC)	0-0-0	P/NP
NO 101#	National Sports Organization (NSS)	0-0-0	P/NP
NS 101#	National Service Scheme (NSS)	0-0-0	P/NP
	Total	**9-4-7**	**34**

Semester II

Course Code	Course Title	Weekly Contact Hours (L-T-P)	Credits
PH 102	Physics – II	2-1-0	6
HS101/ HS 103/ HS 105	Introduction to Philosophy/ Economics/ Reading Literature	3-0-0	6
CS 102 EE 102 ME 102	Abstractions and Paradigms for Programming* Intro. to Elect. And Electronics Circuit* Engineering Mechanics*	3-0-2 2-1-0 2-1-0	8 6 6
MA 102	Linear Algebra and Ordinary Differential Equation – I	3-1-0	8
ME 152	Workshop Practice	0-1-3	5
PH 112	Physics Lab	0-0-3	3
NC 102#	National Cadet Crops (NCC)	0-0-0	P/NP
NO 102#	National Sports Organisation (NSS)	0-0-0	P/NP

NS 102#	National Service Scheme (NSS)	0-0-0	P/NP
	Total	**11/10-3/4-8/6**	**36/34**

Any one of these courses to be taken
* Department Introductory course, specific to the students of concerned Departments

Curriculum of 2st year of BTech in CSE, EE and ME (for AY 2010-11)

2nd Year BTech (Computer Science and Engineering) (For AY 2010-11)

Semester III

Course Code	Subject Name	Weekly Contact Hours (L-T-P)	Credits
HS 111 / HS 113 / HS 115	Introduction to Philosophy / * Economics / Reading Literature	3-0-0 / 3-0-0 / 3-0-0	3 / 3 / 3
MA 201	Mathematics-III (Complex Analysis and Differential Equations-II)	3-1-0	4
EE 104	Basic Electrical and Electronics Engineering	2-1-0	3
CS 201	Discrete Mathematical Structures	2-1-0	3
CS 203	Data Structures and Algorithms	2-1-0	3
CS 253	Data Structures and Algorithms Lab	0-0-3	1.5
CS 261	Program Development and Software Design Lab-I	0-1-4	3
EE 154	Basic Electrical and Electronics Engineering Lab	0-0-2	1
IC 211	Experimental Engineering Lab	0-0-3	1.5
	Total	**12-5-12**	**23**

Semester IV

Course Code	Subject Name	Weekly Contact Hours (L-T-P)	Credits
MA 204	Numerical Methods	3-1-0	4
CS 202	Automata Theory and Logic	2-1-0	3
CS 204	Design and Analysis of Algorithms	2-1-0	3

CS 206	Logic Design	2-1-0	3
CS 208	Software Engineering	2-1-0	3
CS 254	Design and Analysis of Algorithms Lab	0-0-3	1.5
CS 256	Logic Design Lab	0-0-3	1.5
CS 258	Software Engineering Lab	0-0-3	1.5
CS 262	Program Development and Software Design Lab-II	0-1-4	3
	Total	**11-6-13**	**23.5**

*Students have to choose an HSS course other than the one which they have taken in the 2nd Sem.

Curriculum of 1st year BTech (common to all the Departments)

Semester I

Curriculum of 1st Year B. Tech. Program (From AY 2010-11 to AY 2013-14)				**Curriculum of 1st Year B. Tech. Program** (From AY 2014-15 to AY 2018-19)			
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**	**Course Code**	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
CH 103	Chemistry	3-1-0	4	CH 103	Chemistry	3-1-0	4
MA 103	Mathematics-I (Calculus)	3-1-0	4	MA 105	Calculus	3-1-0	4
PH 103	Physics-I (Modern Physics)	2-1-0	3	PH 105	Physics-I	2-1-0	3
HS 107	English Language and Literature	2-0-0	2	HS 159	English Language and Communication	0-3-0	3
CS 103	Computer Programming	2-0-0	2	CS 103	Computer Programming	2-0-0	2
CH 153	Chemistry Lab	0-0-3	1.5	CH 153	Chemistry Lab	0-0-3	1.5
HS 157	English Language Lab	0-0-2	1				
CS 153	Computer Programming Lab	0-0-3	1.5	CS 153 ***(upto AY 2017-18)*** IC 151 ***(from AY 2018-19 onwards)***	Computer Programming Lab	0-0-3	1.5
ME 153	Engineering Graphics	1-0-3	2.5	IC 153	Engineering Graphics	1-0-3	2.5
NC 101/ NO 101/ NS 101	National Cadet Corps (NCC) National Sports Organization (NSO) National Service Scheme (NSS)	0-0-0 0-0-0 0-0-0	P/NP P/NP P/NP	NC 101/ NO 101/ NS 101	National Cadet Corps (NCC) National Sports Organization (NSO) National Service Scheme (NSS)	0-0-0 0-0-0 0-0-0	P/NP P/NP P/NP
	Total	**13-3-11**	**21.5**		**Total**	**11-6-9**	**21.5**

Semester II

Curriculum of 1st Year B. Tech. Program (From AY 2010-11 to AY 2013-14)				Curriculum of 1st Year B. Tech. Program (From AY 2014-15 to AY 2018-19)			
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**	**Course Code**	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
MA 104	Mathematics-II(Linear Algebra and Ordinary Differential Equations-I)	3-1-0	4	MA 106	Linear Algebra and Ordinary Differential Equations-I	3-1-0	4
PH 104	Physics-II (Electricity and Magnetism)	2-1-0	3	PH 106	Physics-II	2-1-0	3
				BSE 102	Bio-Sciences	2-1-0	3
HS 108	Fundamentals of Economics	3-0-0	3	HS 108	Fundamentals of Economics	3-0-0	3
EE 104	Basic Electrical and Electronics Engineering	2-1-0	3	EE 104	Basic Electrical and Electronics Engineering	2-1-0	3
ME 104	Basic Mechanical Engineering	3-0-0	3	ME 106	Basic Mechanical Engineering	2-1-0	3
PH 154	Physics Lab	0-0-3	1.5	PH 156	Physics Lab	0-0-3	1.5
EE 154	Basic Electrical and Electronics Engineering Lab	0-0-2	1	EE 154	Basic Electrical and Electronics Engineering Lab	0-0-2	1
ME 154	Basic Manufacturing Techniques	2-0-2	3	IC 156	Basic Manufacturing Techniques	0-0-3	1.5
NC 102/ NO 102/ NS 102	National Cadet Corps (NCC) National Sports Organization (NSO) National Service Scheme (NSS)	0-0-0 0-0-0 0-0-0	P/NP P/NP P/NP	NC 102 / NO 102 / NS 102	National Cadet Corps (NCC) National Sports Organization (NSO) National Service Scheme (NSS)	0-0-0 0-0-0 0-0-0	P/NP P/NP P/NP
	Total	**15-3-7**	**21.5**		**Total**	**15-4-8**	**23**

Sections and Course structure of 1st year BTech (from AY 2019-20 to AY 2022-23)

Section-A (CSE + CE + MEMS) **Classroom No. 1B-201, Titanium POD**				**Section-B (EE + ME)** **Classroom No. 1D-105, Chromium POD**			
1st (i.e. Autumn) Semester							
Course Code	**Course Title**	**Teaching Hours (L-T-P)**	**Credits**	**Course Code**	**Course Title**	**Teaching Hours (L-T-P)**	**Credits**
CH 103	Chemistry	3-1-0	4	BSE 102	Bio-Sciences	2-1-0	3
MA 105	Calculus	3-1-0	4	MA 105	Calculus	3-1-0	4
PH 105	Physics-I	2-1-0	3	PH 106	Physics-II	2-1-0	3
CS 103	Computer Programming	2-0-0	2	EE 104	Basic Electrical and Electronics Engineering	2-1-0	3
				ME 106	Basic Mechanical Engineering	2-1-0	3
HS 159	English Language and Communication	0-3-0	3	HS 108	Fundamentals of Economics	3-0-0	3
CH 153	Chemistry Lab	0-0-3	1.5	PH 156	Physics Lab	0-0-3	1.5
IC 151	Computer Programming Lab	0-0-3	1.5	EE 154	Basic Electrical and Electronics Engineering Lab	0-0-2	1
IC 153	Engineering Graphics	1-0-3	2.5	IC 156	Basic Manufacturing Techniques	0-0-3	1.5
NO 101	National Sports Organization (NSO)	0-0-0	P/NP	NO 101	National Sports Organization (NSO)	0-0-0	P/NP
	Total	**11-6-9**	**21.5**		**Total**	**14-5-8**	**23**
2nd (i.e. Spring) Semester							
BSE 102	Bio-Sciences	2-1-0	3	CH 103	Chemistry	3-1-0	4
MA 106	Linear Algebra and Ordinary Differential Equations-I	3-1-0	4	MA 106	Linear Algebra and Ordinary Differential Equations-I	3-1-0	4
PH 106	Physics-II	2-1-0	3	PH 105	Physics-I	2-1-0	3
EE 104	Basic Electrical and Electronics Engineering	2-1-0	3	CS 103	Computer Programming	2-0-0	2
ME 106	Basic Mechanical Engineering	2-1-0	3				

HS 108	Fundamentals of Economics	3-0-0	3	HS 159	English Language and Communication	0-3-0	3
EE 154	Basic Electrical and Electronics Engineering Lab	0-0-2	1	IC 151	Computer Programming Lab	0-0-3	1.5
PH 156	Physics Lab	0-0-3	1.5	CH 153	Chemistry Lab	0-0-3	1.5
IC 156	Basic Manufacturing Techniques	0-0-3	1.5	IC 153	Engineering Graphics	1-0-3	2.5
NO 102	National Sports Organization (NSO)		P/NP	NO 102	National Sports Organization (NSO)		P/NP
	Total	**14-5-8**	**23**		**Total**	**11-6-9**	**21.5**

Sections and Course structure of 1st year BTech (from AY 2023-24 onwards)

Section-A (CSE+CE+MEMS+CH+EP)			
Semester-I	**Autumn Semester**		
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
EE 101	Basic Electrical Engineering	1-1-0	2
ME 101	Engineering Mechanics	2-0-0	2
PH 107	Basics of Physics	2-1-0	3
PH 157	Physics Lab- I	0-0-2	1
MA 101N	Calculus-I (half Semester)	3-1-0 (=4/2)	2
MA 103N	Calculus-II (half Semester)	3-1-0 (=4/2)	2
HS 109	Language and Composition	2-0-0	2
HS XXX	Flexible Elective (HSS)	1-0-0	1
IC 152	Makerspace	1-0-6	4
CS 103	Computer Programming	2-0-0	2
IC 151	Computer Programming Lab	0-0-3	1.5
NO 101	National Sports Organization (NSO)	0-0-0	P/NP
	Total	**14-3-11**	**22.5**

Section-B (EE+ME+MC+SSE)			
Semester-I	**Autumn Semester**		
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
EE 101	Basic Electrical Engineering	1-1-0	2
ME 101	Engineering Mechanics	2-0-0	2
CH 105	Chemistry	3-0-0	3
CH 155	Chemistry Lab	0-0-2	1
MA 101N	Calculus-I (half Semester)	3-1-0 (=4/2)	2
MA 103N	Calculus-II (half Semester)	3-1-0 (=4/2)	2
HS 109	Language and Composition	2-0-0	2
HS XXX	Flexible Elective (HSS)	1-0-0	1
CS 103	Computer Programming	2-0-0	2
IC 151	Computer Programming Lab	0-0-3	1.5
NO 101	National Sports Organization (NSO)	0-0-0	P/NP
	Total	**14-2-5**	**18.5**

Semester-II	Spring Semester		
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
BSE 102	Biosciences	2-1-0	3
MA 102N	Linear Algebra (half Semester)	2-1-0 (=3/2)	1.5
MA 104N	Differential Equations-I (half Semester)	2-1-0 (=3/2)	1.5
ES 102	Environmental Studies: Scientific and Engineering Aspects (half Semester)	2-1-0 (=3/2)	1.5
HS 102	Environmental Studies: Social Aspects (half Semester)	2-1-0 (=3/2)	1.5
HS 104	Fundamentals of Economics	2-0-0	2
CH 105	Chemistry	3-0-0	3
CH 155	Chemistry Lab	0-0-2	1
ZZ XXX	Flexible Elective	1-0-0	1
ZZ XXX	Flexible Elective	1-0-0	1
ZZ XXX	Flexible Elective (HSS)	1-0-0	1
NO 102	National Sports Organization (NSO)	0-0-0	P/NP

Semester-II	Spring Semester		
Course Code	**Course Title**	**Weekly Contact Hours (L-T-P)**	**Credits**
BSE 102	Biosciences	2-1-0	3
MA 102N	Linear Algebra (half Semester)	2-1-0 (=3/2)	1.5
MA 104N	Differential Equations-I (half Semester)	2-1-0 (=3/2)	1.5
ES 102	Environmental Studies: Scientific and Engineering Aspects (half Semester)	2-1-0 (=3/2)	1.5
HS 102	Environmental Studies: Social Aspects (half Semester)	2-1-0 (=3/2)	1.5
HS 104	Fundamentals of Economics	2-0-0	2
IC 152	Makerspace	1-0-6	4
PH 107	Basics of Physics	2-1-0	3
PH 157	Physics Lab- I	0-0-2	1
ZZ XXX	Flexible Elective	1-0-0	1
ZZ XXX	Flexible Elective	1-0-0	1
ZZ XXX	Flexible Elective (HSS)	1-0-0	1

	Total	**14-3-2**	**18**

NO 102	National Sports Organization (NSO)	0-0-0	P/NP
	Total	**14-4-8**	**22**

Curriculum for BTech (CSE)

Semester III

Curriculum of 2nd Year B. Tech. (CSE) (From AY 2011-12 to AY 2013-14)					**Curriculum of 2nd Year B. Tech. (CSE)** (From AY 2014-15 onwards to AY 2023-24)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**		**Course Code**	**Course Title**	**Weekly L-T-P**	**Credits**
HS 201 / HS 203 / HS 205 HS 207	Understanding Philosophy / Psychology / Sociology / French Language – I	3-0-0 / 3-0-0 / 2-1-0 / 2-1-0	3 / 3 / 3 / 3		ZZ XXX	Course-I for Minor Program	X-X-X	3
MA 201	Mathematics-III (Complex Analysis and Differential Equations-II)	3-1-0	4		MA 203	Complex Analysis and Differential Equations-II	3-1-0	4
CS 201	Discrete Mathematical Structures	2-1-0	3		CS 201	Discrete Mathematical Structures	2-1-0	3
CS 203	Data Structures and Algorithms	2-1-0	3		CS 203	Data Structures and Algorithms	2-1-0	3
CS 205	Abstraction and Paradigms for Programming	2-1-0	3		CS 207	Data Base & Information Systems	**3-0-0**	**3**
CS 253	Data Structures and Algorithms Lab	0-0-3	1.5		CS 253	Data Structures and Algorithms Lab	0-0-3	1.5
CS 255	Abstraction and Paradigms for Programming Lab	0-0-3	1.5		CS 257	Data Base & Information Systems Lab	**0-0-3**	**1.5**
IC 211	Experimental Engineering Lab	0-0-3	1.5		IC 211	Experimental Engineering Lab	0-0-3	1.5
	Total	**12/11-4/5-9**	**20.5**			**Total**	**10-3-9**	**17.5 / 20.5**

Go to Index

Curriculum of 2nd Year B. Tech. (CSE) (From AY 2024-25 onwards)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
ZZ xxx	Course-I for Minor Program	x-x-x	3
MA 205	Complex Analysis	3-1-0 (1/2 Sem)	2
MA 207	Differential Equations-II	3-1-0 (1/2 Sem)	2
CS 201	Discrete Mathematical Structures	2-1-0	3
CS 203	Data Structures and Algorithms	2-1-0	3
CS 207N	Database and Information Systems	2-1-0	3
CS 209	Logic Design	2-0-2	3
CS 253	Data Structures and Algorithms Lab	0-0-3	1.5
CS 257	Database and Information Systems Lab	0-0-3	1.5
CS 2XX	Department Elective I	x-x-x	3
	Total	11-4-8 **	22/25

Semester IV

Curriculum of 2nd Year B. Tech. (CSE) (From AY 2011-12 to AY 2013-14)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
HS 208	French Language – II +	2-1-0	3
MA 204	Numerical Methods	3-1-0	4
CS 202	Automata Theory and Logic	2-1-0	3

Curriculum of 2nd Year B. Tech. (CSE) (From AY 2014-15 onwards to AY 2023-24)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
ZZ XXX	Course-II for Minor Program	X-X-X	3
MA 204	Numerical Methods	3-0-2	4
CS 202	Automata Theory and Logic	2-1-0	3

CS 204	Design and Analysis of Algorithms	2-1-0	3		CS 204	Design and Analysis of Algorithms	2-1-0	3
CS 206	Logic Design	2-1-0	3		CS 206	Logic Design	2-1-0	3
CS 208	Software Engineering	2-1-0	3		CS 208	Software Engineering	2-1-0	3
CS 254	Design and Analysis of Algorithms Lab	0-0-3	1.5		CS 254	Design and Analysis of Algorithms Lab	0-0-3	1.5
CS 256	Logic Design Lab	0-0-3	1.5		CS 256	Logic Design Lab	0-0-3	1.5
CS 258	Software Engineering Lab	0-0-3	1.5		CS 258	Software Engineering Lab	0-0-3	1.5
	Total	**11/13-5/6-9**	**20.5 / 23.5**			**Total**	**11-5-9**	**20.5 / 23.5**

+ Additional course ONLY for those students who have taken and passed HS 207 in their 3rd Semester.

Curriculum of 2nd Year B. Tech. (CSE)
(From AY 2024-25 onwards) (Batch admitted in and after AY 2023-24)

Course Code	Course Title	Weekly L-T-P	Credits
ZZ xxx	Course-II for Minor Program	x-x-x	3
MA 204N	Numerical Methods	2-0-2	3
CS 202	Automata Theory and Logic	2-1-0	3
CS 204	Design and Analysis of Algorithms	2-1-0	3
CS 210	Computer Architecture	2-1-0	3
CS 254	Design and Analysis of Algorithms Lab	0-0-3	1.5
CS 264	Computer Architecture Lab	0-0-3	1.5
CS 2XX	Department Elective II	x-x-x	3
ZZ 2XX	Institute Elective-I	x-x-x	3
	Total	**8-3-8**	**21/24**

Curriculum for BTech (CSE)

Semester V

Curriculum of 3rd Year B. Tech. (CSE) (From AY 2011-12 to AY 2014-15)					**Curriculum of 3rdYear B. Tech. (CSE)** (From AY 2015-16 to AY 2024-25)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**		**Course Code**	**Course Title**	**Weekly L-T-P**	**Credits**
HS xxx	HSS Course	3-0-0	3		ZZ XXX	Course-III for Minor Program	X-X-X	3
CS 301	Data Base & Information Systems	3-0-0	3		**CS 309**	**Parallel Computing**	**2-1-0**	**3**
CS 303	Operating Systems	2-1-0	3		CS 303	Operating Systems	2-1-0	3
CS 305	Computer Architecture	2-1-0	3		CS 305	Computer Architecture	2-1-0	3
CS 307	Optimization Algorithms and Techniques	3-0-0	3		CS 307	Optimization Algorithms and Techniques	2-1-0	3
CS 351	Data Base & Information Systems Lab	0-0-3	1.5		**CS 359**	**Parallel Computing Lab**	**0-0-3**	**1.5**
CS 353	Operating Systems Lab	0-0-3	1.5		CS 353	Operating Systems Lab	0-0-3	1.5
CS 355	Computer Architecture Lab	0-0-3	1.5		CS 355	Computer Architecture Lab	0-0-3	1.5
CS 357	Optimization Algorithms and Techniques Lab	0-0-3	1.5		CS 357	Optimization Algorithms and Techniques Lab	0-0-3	1.5
	Total	**13-2-12**	**21**			**Total**	**8-4-12**	**18 / 21**

Curriculum of 3rd Year B. Tech. (CSE) (From AY 2025-26 onwards) (Batch admitted in and after AY 2023-24)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
ZZ xxx	Course III - Minor Program	x-x-x	3
CS 311	Parallel Computing	2-0-2	3
CS 303	Operating Systems	2-1-0	3
CS 307	Optimization Algorithms and Techniques	2-1-0	3

CS 313	Computer Networks	2-0-2	3
CS 353N	Operating Systems Lab	0-0-2	1
CS 357N	Optimization Algorithms and Techniques Lab	0-0-2	1
CS 3XX	Department Elective III	x-x-x	3
ZZ 3XX	Institute Elective II	x-x-x	3
	Total	**8-2-8**	**20/23**

Semester VI

Curriculum of 3rd Year B. Tech. (CSE) (From AY 2011-12 to AY 2014-15)				**Curriculum of 3rd Year B. Tech. (CSE)** (From AY 2015-16 to AY 2024-25)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**	**Course Code**	**Course Title**	**Weekly L-T-P**	**Credits**
HS 302	Environmental Studies: Social Aspects	3-0-0	1.5	HS 302	Environmental Studies: Social Aspects	3-0-0	1.5
ES 302	(Half Semester course) Environmental Studies: Scientific and Engineering Aspects (Half Semester course)	3-0-0	1.5	ES 302	(Half Semester course) Environmental Studies: Scientific and Engineering Aspects (Half Semester course)	3-0-0	1.5
CS 302	Computer Graphics and Visualization	3-0-0	3	CS 302	Computer Graphics and Visualization	2-1-0	3
CS 304	Artificial Intelligence	3-0-0	3	CS 304N	Computational Intelligence	2-1-0	3
CS 306	Computer Networks	3-0-0	3	CS 306	Computer Networks	2-1-0	3
CS 308	Compiler Techniques	3-0-0	3	CS 308	Compiler Techniques	2-1-0	3
CS 352	Computer Graphics and Visualization Lab	0-0-3	1.5	CS 352	Computer Graphics and Visualization Lab	0-0-3	1.5
CS 354	Artificial Intelligence Lab	0-0-3	1.5	CS 354N	Computational Intelligence Lab	0-0-3	1.5
CS 356	Computer Networks Lab	0-0-3	1.5	CS 356	Computer Networks Lab	0-0-3	1.5
CS 358	Compiler Techniques Lab	0-0-3	1.5	CS 358	Compiler Techniques Lab	0-0-3	1.5

CS 391	Summer Internship (After the completion of the 6th semester)							
	Total	**15-0-12**	**21**			**Total**	**11-4-12**	**21**

Curriculum of 3rd Year B. Tech. (CSE) (From AY 2025-26 onwards) (Batch admitted in and after AY 2023-24)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
ZZ xxx	Course IV - Minor Program	x-x-x	3
CS 302N	Computer Graphics and Visualization	2-0-2	3
CS 304N	Computational Intelligence	2-1-0	3
CS 310	Software Engineering	2-0-2	3
CS 308N	Compiler Techniques	2-0-2	3
CS 354N	Computational Intelligence Lab	0-0-3	1.5
CS 3XX	Department Elective IV	x-x-x	3
CS 3XX	Department Elective V	x-x-x	3
ZZ 3XX	Institute Elective III	x-x-x	3
	Total	**8-1-9**	**22.5/25.5**

Curriculum for BTech (CSE)

Semester VII

Curriculum of 4th Year B. Tech. (CSE) (From AY 2011-12 to AY 2013-14)					**Curriculum of 4th Year B. Tech. (CSE)** (From AY 2014-15 to AY 2025-26)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**		**Course Code**	**Course Title**	**Weekly**	**Credits**

							L-T-P	
CS 401	Soft Computing	3-0-0	3		CS 493	**B Tech Project (BTP)** 1. Student can do BTech project either outside the institute or within the institute under a supervision of an IIT Indore Faculty. 2. Summer Internship, if any, will be part of B Tech Project. 3. The choice is to be made latest by 30th April. 4. Duration: 6-7 months during 2nd week of May to Last week of Nov. 5. Last Date of Thesis submission: 1st week of Dec. 6. Last Date of Submission of Grades: 2nd week of Dec.	0-0-40	20
CS xxx	Department Elective – I	x-x-x	3					
CS xxx	Department Elective – II	x-x-x	3					
XX xxx	Institute Elective – I	x-x-x	3					
CS 451	Soft Computing Lab	0-0-3	1.5					
CS 491	B.Tech. Project (Stage 1)	0-0-12	6					
CS 391	Evaluation of Summer Internship	0-2-0	2					
		Total	**21.5**				**Total**	**20**

Curriculum of 4th Year B. Tech. (CSE)
(From AY 2026-27 onwards) (Batch admitted in and after AY 2023-24)

Course Code	Course Code	Course Code	Credits
ZZ XXX	Course-V for Minor project/field study/white paper/domain comprehension (Seminar)/Lab course	x-x-x	2
ZZ 493N	B. Tech Project (BTP)	0-0-32	16
ZZ 495 ZZ XXX	Internship **OR** Professional/ Societal-Connect basket course	x-x-x	1.5

Go to Index

Total		**17.5/19.5**

Semester VIII

Curriculum of 4th Year B. Tech. (CSE) (From AY 2011-12 to AY 2013-14)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
CS 402	Parallel Computing	3-0-0	3
CS 452	Parallel Computing Lab	0-0-3	1.5
CS xxx	Department Elective - III	x-x-x	3
CS xxx	Department Elective - IV	x-x-x	3
XX xxx	Institute Elective – II	x-x-x	3
CS 492	B. Tech. Project (Stage 2)	0-0-12	6
		Total	**19.5**

Curriculum of 4th Year B. Tech. (CSE) (From AY 2014-15 to AY 2015-16)			
Course Code	**Course Title**	**Weekly L-T-P**	**Credits**
CS 401 / CS 601	Soft Computing %	2-0-2	3
CS 402	Parallel Computing %	2-0-2	3
ZZ xxx	Elective-I	x-x-x	3
ZZ xxx	Elective-II	x-x-x	3
ZZ xxx	Elective-III	x-x-x	3
ZZ xxx	Elective-IV	x-x-x	3
		Total	**18**
From AY 2016-17 to AY 2019-20			
CS 419 / ICS 419	Computer Vision	2-1-0	3
ZZ xxx	Elective-I	x-x-x	3
ZZ xxx	Elective-II	x-x-x	3
ZZ xxx	Elective-III	x-x-x	3
ZZ xxx	Elective-IV (or Course-IV for Minor Program)	x-x-x	3
ZZ xxx	Elective-V (or Course-V for Minor Program)	x-x-x	3
		Total	18
From AY 2020-21 to AY 2025-26			
CS 419 / ICS 419	Computer Vision	2-1-0	3

		CS xxx	Department Elective-I	x-x-x	3
		CS xxx	Department Elective-II	x-x-x	3
		CS xxx	Department Elective-III	x-x-x	3
		ZZ xxx	Open Elective-I (or Course-IV for Minor Program)	x-x-x	3
		ZZ xxx	Open Elective-II (or Course-V for Minor Program)	x-x-x	3
				Total	**18**

Curriculum of 4th Year B. Tech. (CSE)
(From AY 2026-27 onwards) (Batch admitted in and after AY 2023-24)

Course Code	Course Code	Course Code	Credits
CS 4XX	Department Elective VI	x-x-x	3
CS 4XX	Department Elective VII	x-x-x	3
ZZ 4XX	Institute Elective IV	x-x-x	3
ZZ 4XX	Institute Elective V	x-x-x	3
ZZ 4XX	Institute Elective VI	x-x-x	3
	Total	**X-X-X**	**15**

CSE courses available for the Elective Courses of BTech Program in CSE

CS 211 : UX/UI Design (2-0-2-3)
CS 213 : Matrix Factorization and Applications (2-1-0-3)
CS 215 : Mathematics for AI and ML (2-1-0 -3)
CS 212 : Foundation of Algebraic Graph Theory (2-1-0-3)
CS 214 : Foundations of Hardware Security (2-1-0-3)
CS 216 : Introduction to Blockchain (2-1-0-3) – ½ semester
CS 218 : Programmable and Interoperable Blockchains (2-1-0-3) – ½ semester
CS 315 : Introduction to Complexity Theory (2-1-0 3)

CS 317 : Introduction to Internet of Things (2-1-0-3)
CS 319 : Foundations of Cryptography (2-1-0-3)
CS 321 : Introduction to Big Data Analysis (2-1-0-3)
CS 312 : Foundations of Secure Computation (2-1-0-3)
CS 314 : Computer and Network Security (2-1-0-3)
CS 401 / CS 601 : Soft Computing (2-0-2-3)
CS 403/ CS 603 : Machine Learning (2-0-2-3)
CS 404 / EE 304 : Digital Signal Processing (3-1-0-4)
CS 406 / CS 606 : Data Mining and Data Warehousing (2-0-2-3)
CS 407 : Peripherals and Interfaces (2-0-2-3)
CS 408 : Algorithms for Convex Programming (2-0-2-3)
CS 409 / CS 609 : Advanced Topics in Database Management Systems (2-1-0-3)
CS 410 : Genetic Algorithms (2-0-2-3)
CS 411/ CS 611 : Advanced Algorithms (2-0-2-3)
CS 412/ CS 612 : Pattern Recognition (2-0-2-3)
CS 413 : Topics in Artificial Intelligence Programming (2-1-0-3)
CS 414 / CS 614 : Cloud Computing and Applications (2-1-0-3)
CS 416 / CS 616 : Service Oriented Systems (2-1-0-3)
CS 417 / CS 617 : Cryptography and Network Security (2-0-2-3)
CS 418 / CS 618 : Systems and Usable Security (2-1-0-3)
CS 419 / ICS 419 / CS 619: Computer Vision (2-1-0-3) [From AY 2016-17 onward, it will be a compulsory course]
CS 420 / CS 620 : Embedded Systems (2-1-0-3)
CS 422 / CS 622 : Numerical Simulation (2-1-0-3)
CS 424 : Functional and Logic Programming (2-0-2-3)
CS 425 / CS 625 : Natural Language Processing (2-0-2-3)
CS 426/ CS 626 : Foundations of Cyber-Physical Systems (2-0-2-3)
CS 427/ CS 627 : Advanced Computer Networks (2-1-0-3)
CS 428/ CS 628 : Algorithmic Graph Theory (2-1-0-3)
CS 430/ CS 630 : Data Center Networking (2-1-0-3)
CS 432/ CS 632 : Reinforcement Learning (2-0-2-3)

CS 334/ CS 434/ CS 634: Wireless Networks and Applications (2-1-0-3)
CS 435/ CS 635 : Deep Learning (2-0-2-3)
CS 438/ CS 638 : Network Softwarization and Management (2-0-2-3)
CS 440/640 : Distributed Network Algorithms (2-1-0-3)
CS 442/642 : Algebraic Graph Theory
CS 444 : Advanced Blockchain (2-1-0-3)
CS 446 : Blockchain for Responsible Computing (2-1-0-3)
CS 448/648 : Software Verification and Certification (2-1-0-3)
CS 450/CS 650 : Generative AI (2-1-0-3)
CS 454/ CS 654 : Distributed Algorithms for Robots and Agents (2-1-0-3)
CS 456/ CS 656 : Software-defined IoT (2 - 0 - 2 – 3)

Course Structure for the Interdisciplinary Dual Degree Program at the Department of Computer Science and Engineering
(From AY 2025-26 onwards)
{Senate *resolution no.54.8.}*

Scope of the program: The students, from any department apart from the Computer Science and Engineering (CSE) and Mathematics and Computing (MnC) departments, have to take the FOUR core courses as prescribed till the 6th semester in CSE along with the regular BTech degree in their respective engineering discipline. From the 7th semester onwards, s/he can pursue an interdisciplinary degree in CSE and register for the MTech in CSE.

Minimum Education Qualification (MEQ):

- Completed prerequisite courses up to six semesters with a CPI of at least 8.5
- The students will be shortlisted based on their CPI after the sixth semester.

Intake: 30 per batch (15 Mathematics and Computing, Computer Science and Engineering. 15 seats from other departments)

Total Credits: 67 (excluding 12 extra credits as a prerequisite)

Year	Semester	Course Code	Course	Weekly L-T-P	Credits
Years II and III	Sem 3-6	CS 203D	Data Structures and Algorithms	2-1-0	3
		CS 204D	Design and Analysis of Algorithms	2-1-0	3
		CS 201D	Discrete Mathematical Structures	2-1-0	3
		CS 202D	Automata Theory and Logic	2-1-0	3
Year IV	Sem 7	CS 639	Computing Foundations: Operating Systems	1-0-2	2

		CS 641	Computing Foundations: Compiler Design		2-0-2	1.5 (3/2)
		CS 643	Computing Foundations: Computer Architecture		2-0-2	1.5 (3/2)
		CS 411/611	Advanced Algorithms		2-1-0	3
		ZZ XXX	Interdisciplinary Research Project-I		0-0-8	4
	Sem 8	ZZ XXX	Interdisciplinary Research Project-II		0-0-8	4
		ZZ 6XX	Elective-1		X-X-X	3
		ZZ 6XX	Elective-2		X-X-X	3
		CS 4XX/6XX	Elective-3	Bucket -1 (AI/ML) Bucket -2 (Networking and Cyber Security)	X-X-X	3
		CS 4XX/6XX	Elective-4		X-X-X	3
		CS 4XX/6XX	Elective-5		X-X-X	3
Year V	Sem 9	CS 799	M. Tech. Research Project (Stage I)		0-0-36	18
	Sem 10	CS 800	M. Tech. Research Project (Stage-II)		0-0-36	18
Total						67

Structure of the Minor programs [from AY 2014-15 to AY 2020-21]

A student has to register and pass at least FIVE courses (three core courses and two elective courses) as prescribed for a minor program in order to get a minor degree in that specialization along with the regular BTech degree in his/her engineering Department. A minor program will run only when at least TEN students register for it. Following minor programs are available from AY 2014-15 onwards.

1. Minor program in Biosciences and Biomedical Engineering (BSBE): To get a minor degree in BSBE, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course*** BSE 101 Bio-Sciences for successful minor degree in BSBE.

2. MINOR PROGRAM IN CHEMISTRY: To get a minor degree in Chemistry, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course CH 103.*** Following are courses for successful minor degree in Chemistry.

3. Minor Program in HSS: A student needs to register and pass **at least FIVE prescribed courses of Humanities and Social Sciences** ***excluding the core courses*** HS 159 and HS 108 for successful minor degree in Humanities or Social Sciences.

4. **Minor Program in Astronomy (from AY 2016-17):** To get a minor degree in Astronomy, a student needs to register and pass **at least FIVE prescribed** courses***.*** Following are courses for successful minor degree in Astronomy.

Course structures of various Minor programs

Semester: Minor course	**Minor Program in BSBE**	**Minor Program in Chemistry**	**Minor Program in Humanities and Social Sciences**	**Minor Program in Astronomy (from AY 2016-17 onwards)**
3rd : Minor1	BSE 201: Biophysics	CH 201: Molecules that Change the World	HS 201: Understanding Philosophy HS 203: Psychology HS 205: Sociology HS 207: French Language-I	AA 201: Introduction to Astronomy
4th : Minor 2	BSE 202: Biomedical Technologies	CH 202: Chemistry of Transition Metals and Lanthanides &	HS 206: Paradigms and Turning Points # HS 208: French Language-II HS 210: Indian Economy HS 211: German Literature and Culture Studies HS 214: History of Indian Culture and Civilization HS 216: Introduction to Hindi Cinema	AA 202N: Astronomical Techniques
5th : Minor 3	BSE 301: Introduction to Molecular Biology	CH 301: Functional Materials	HS 311: Life and Thought of Gandhi HS 313: History of Early Cinema	AA 301: High Energy Astronomy

			HS 315: Sociology of Science and Technology HS 323: International Economics HS 341: Appreciating Indian English Literature	
8th : Two elective courses as Minor 4 and Minor 5	BSE 402: Cancer Diagnosis and Therapy BSE 404/ BSE 604: Biomedical Imaging BSE 405/ BSE 605: Molecular Biophysics BSE 413/ BSE 613: Omics Technologies BSE 417/ BSE 617: Biomolecular Modeling EE 419/ EE 619: Biomedical Optics ME 407/ME 607: Bio-fluid Mechanics	CH 402: Chemistry in Industry CH 404: Chemical Physics CH 406: Nuclear Science	IHS 402: Twentieth Century World History: Critical Perspectives HS 412/ 612: Contemporary Indian Thought HS 418/ 618: Sustainability Studies HS 424/ HS 624: Econometrics-I IHS 422 / HS 622: Development Economics IHS 425: Money and Banking HS 426: Economics of Innovation HS 442/ HS 642: Language and Mind IHS 443/ HS 643: Contemporary Short Fiction IHS 444: Literature of the Twentieth Century IHS 482: Introduction to International Development and Area Studies	AA 404/ AA 604: Spacecraft and Payload Attitude Dynamics, Control and Pointing AA 471N/ AA 671N: Relativity and Cosmology AA 472N/ AA 672N: Galactic and Extragalactic Astronomy AA 474 / AA 674: Basics of Radio Astronomy AA 476/ AA 676: Satellite Based Navigation Systems AA 478/ AA 678: Space Weather

& A student who takes CH 202 will not be allowed to take ME 416/616 in his/her 8th Semester

Structure of the Minor programs [For AY 2021-22]

A student has to register and pass at least FIVE courses (three core courses and two elective courses) as prescribed for a minor program in order to get a minor degree in that specialization along with the regular BTech degree in his/her engineering Department. A minor program will run only when at least TEN students register for it. Following minor programs are available from AY 2014-15 onwards.

1. Minor program in Biosciences and Biomedical Engineering (BSBE): To get a minor degree in BSBE, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course*** BSE 101 Bio-Sciences for successful minor degree in BSBE.

2. MINOR PROGRAM IN CHEMISTRY: To get a minor degree in Chemistry, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course CH 103.*** Following are courses for successful minor degree in Chemistry.

3. Minor Program in HSS: A student needs to register and pass **at least FIVE prescribed courses of Humanities and Social Sciences** ***excluding the core courses*** HS 159 and HS 108 for successful minor degree in Humanities or Social Sciences.

4. **Minor Program in Astronomy (from AY 2016-17):** To get a minor degree in Astronomy, a student needs to register and pass **at least FIVE prescribed** courses***.*** Following are courses for successful minor degree in Astronomy.

Course structures of various Minor programs

Semester: Minor course	**Minor Program in BSBE**	**Minor Program in Chemistry**	**Minor Program in Humanities and Social Sciences**	**Minor Program in Astronomy (from AY 2016-17 to AY 2021-22)**
3rd : Minor1	BSE 201: Biophysics	CH 201: Molecules that Change the World	HS 201: Understanding Philosophy HS 203: Psychology HS 205: Sociology HS 207: French Language-I	AA 201: Introduction to Astronomy
4th : Minor 2	BSE 202: Biomedical Technologies	CH 202: Chemistry of Transition Metals and Lanthanides &	HS 206: Paradigms and Turning Points # HS 208: French Language-II HS 210: Indian Economy HS 211: German Literature and Culture Studies HS 214: History of Indian Culture and Civilization HS 216: Introduction to Hindi Cinema	AA 202N: Astronomical Techniques AA 204: Introduction to Space Exploration
5th : Minor 3	BSE 301: Introduction to	CH 301: Functional	HS 311: Life and Thought of Gandhi	AA 301: High Energy

	Molecular Biology	Materials	HS 313: History of Early Cinema HS 315: Sociology of Science and Technology HS 323: International Economics HS 341: Appreciating Indian English Literature	Astronomy AA 303: IoT for Space Applications
8th : Two elective courses as Minor 4 and Minor 5	BSE 402: Cancer Diagnosis and Therapy BSE 404/ BSE 604: Biomedical Imaging BSE 405/ BSE 605: Molecular Biophysics BSE 413/ BSE 613: Omics Technologies BSE 417/ BSE 617: Biomolecular Modeling BSE 419/ BSE 619: Renewable Energy Technologies EE 419/ EE 619: Biomedical Optics ME 407/ME 607: Bio-fluid Mechanics	CH 402: Chemistry in Industry CH 404: Chemical Physics CH 406: Nuclear Science	IHS 402: Twentieth Century World History: Critical Perspectives HS 412/ 612: Contemporary Indian Thought HS 418/ 618: Sustainability Studies HS 424/ HS 624: Econometrics-I IHS 422 / HS 622: Development Economics IHS 425: Money and Banking HS 426: Economics of Innovation HS 442/ HS 642: Language and Mind IHS 443/ HS 643: Contemporary Short Fiction IHS 444: Literature of the Twentieth Century IHS 482: Introduction to International Development and Area Studies	AA 404/ AA 604: Spacecraft and Payload Attitude Dynamics, Control and Pointing AA 471N/ AA 671N: Relativity and Cosmology AA 472N/ AA 672N: Galactic and Extragalactic Astronomy AA 474 / AA 674: Basics of Radio Astronomy AA 476/ AA 676: Satellite Based Navigation Systems AA 478/ AA 678: Space Weather

& *A student who takes CH 202 will not be allowed to take ME 416/616 in his/her 8th Semester*

Structure of the Minor programs [from AY 2022-23 onwards]

A student has to register and pass at least FIVE courses (three core courses and two elective courses) as prescribed for a minor program in order to get a minor degree in that specialization along with the regular BTech degree in his/her engineering Department. A minor program will run only when at least TEN students register for it. Following minor programs are available from AY 2014-15 onwards.

1. Minor program in Biosciences and Biomedical Engineering (BSBE): To get a minor degree in BSBE, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course*** BSE 101 Bio-Sciences for successful minor degree in BSBE.

2. MINOR PROGRAM IN CHEMISTRY: To get a minor degree in Chemistry, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course CH 103.*** Following are courses for successful minor degree in Chemistry.

3. Minor Program in Economics: A student needs to register and pass **at least FIVE prescribed courses of Humanities and Social Sciences** ***excluding the core courses*** HS 159 and HS 108 for successful minor degree in Humanities or Social Sciences.

4. **Minor Program in Liberal Arts**

5. **Minor Program in Astronomy and Space Engineering (from AY 2022-23):** To get a minor degree in Astronomy, a student needs to register and pass **at least FIVE prescribed** courses**.** Following are courses for successful minor degree in Astronomy.

Course structures of various Minor programs

Semester: Minor course	**Minor Program in BSBE**	**Minor Program in Chemistry**	**Minor Program in Economics (from AY 2022-23 onwards with BTech batch admitted in AY 2021-22)**	**Minor Program in Liberal Arts (from AY 2022-23 onwards with BTech batch admitted in AY 2021-22)**	**Minor Program in Astronomy (from AY 2016-17 to AY 2021-22) Minor Program in Astronomy and Space Engineering (from AY 2022-23 onwards with BTech batch admitted in AY 2021-22)**
3rd : Minor1	BSE 201: Biophysics	CH 201: Molecules that Change the World	HS 209: Intermediate Microeconomics	HS 201: Understanding Philosophy HS 203: Psychology HS 205: Sociology HS 207: French Language-I Psychology	AA 201: Introduction to Astronomy

4th : Minor 2	BSE 202: Biomedical Technologies	CH 202: Chemistry of Transition Metals and Lanthanides &	HS 210: Indian Economy	HS 206: Paradigms and Turning Points HS 208: French Language-II HS 211: German Literature and Culture Studies HS 212: History of India after Independence, 1947-2000 HS 213: Cognitive HS 214: History of Indian Culture and Civilization HS 216: Introduction to Hindi Cinema	AA 202N: Astronomical Techniques AA 204: Introduction to Space Exploration
5th : Minor 3	BSE 301: Introduction to Molecular Biology	CH 301: Functional Materials	HS 323: International Economics HS 325: Industrial Organization	HS 311: Life and Thought of Gandhi HS 313: History of Early Cinema HS 315: Sociology of Science and Technology HS 321: History of Modern Indian Business HS 327: Mind, Action, and Technology HS 341: Appreciating Indian English Literature	AA 301: High Energy Astronomy AA 303: IoT for Space Applications

8th : Two elective courses as Minor 4 and Minor 5	BSE 402: Cancer Diagnosis and Therapy BSE 404/ BSE 604: Biomedical Imaging BSE 405/ BSE 605: Molecular Biophysics BSE 413/ BSE 613: Omics Technologies BSE 417/ BSE 617: Biomolecular Modeling BSE 419/ BSE 619: Renewable Energy Technologies EE 419/ EE 619: Biomedical Optics ME 407/ME 607: Bio-fluid Mechanics	CH 402: Chemistry in Industry CH 404: Chemical Physics CH 406: Nuclear Science	HS 418/ 618: Sustainability Studies IHS 422 / HS 622: Development Economics IHS 425: Money and Banking HS 426: Economics of Innovation HS 424/ HS 624 Econometrics-I HS 431: Innovation and Intellectual Property Rights	IHS 402: Twentieth Century World History: Critical Perspectives HS 412/ 612: Contemporary Indian Thought HS 442/ HS 642: Language and Mind IHS 443/ HS 643: Contemporary Short Fiction IHS 444: Literature of the Twentieth Century IHS 482: Introduction to International Development and Area Studies HS 446: Music and Literary Modernism HS 481/ HS 681: Language, Mind and Society	AA 404/ AA 604: Spacecraft and Payload Attitude Dynamics, Control and Pointing AA 410/ AA 410: Spatial Informatics AA 412/ AA 612: Microwave Remote Sensing AA 471N/ AA 671N: Relativity and Cosmology AA 472N/ AA 672N: Galactic and Extragalactic Astronomy AA 474 / AA 674: Basics of Radio Astronomy AA 476/ AA 676: Satellite Based Navigation Systems AA 478/ AA 678: Space Weather

Structure of the Minor programs for AY 2024-25 onwards (For all UG batches admitted in and after AY 2023-24)

A student has to register and pass at least FIVE courses (three core courses and two elective courses) as prescribed for a minor program in order to get a minor degree in that specialization along with the regular BTech degree in his/her engineering Department. A minor program will run only when at least TEN students register for it. Following minor programs are available from AY 2014-15 onwards.

1. Minor program in Biosciences and Biomedical Engineering (BSBE): To get a minor degree in BSBE, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course*** BSE 101 Bio-Sciences for successful minor degree in BSBE.

2. MINOR PROGRAM IN CHEMISTRY: To get a minor degree in Chemistry, a student needs to register and pass **at least FIVE prescribed** courses ***excluding the core course CH 103.*** Following are courses for successful minor degree in Chemistry.

3. Minor Program in Economics: A student needs to register and pass **at least FIVE prescribed courses of Humanities and Social Sciences** ***excluding the core courses*** HS 159 and HS 108 for successful minor degree in Humanities or Social Sciences.

4. **Minor Program in Liberal Arts**

5. **Minor Program in Astronomy and Space Engineering (from AY 2022-23):** To get a minor degree in Astronomy, a student needs to register and pass **at least FIVE prescribed** courses**.** Following are courses for successful minor degree in Astronomy.

Course structures of various Minor programs

Semester: Minor course	**Minor Program in BSBE**	**Minor Program in Chemistry**	**Minor Program in Economics From AY 2024-25 (Batch Admitted in and after AY 2023-24)**	**Minor Program in Liberal Arts From AY 2024-25 (Batch Admitted in and after AY 2023-24)**	**Minor Program in Astronomy From AY 2024-25 (Batch Admitted in and after AY 2023-24)**
3rd : Minor1	BSE 201: Biophysics	CH 201: Molecules that Change the World	HS 209: Intermediate Microeconomics	HS 211: German Literature and Culture Studies HS 212: History of India after Independence, 1947-2000 HS 203: Psychology	AA 201: Introduction to Astronomy

				HS 205: Sociology HS 221 Fundamentals of Linguistics HS 223 Language Variation: Culture and Society	
4th : Minor 2	BSE 202: Biomedical Technologies	CH 202: Chemistry of Transition Metals and Lanthanides &	HS 210: Indian Economy	HS 206: Paradigms and Turning Points HS 214: History of Indian Culture and Civilization HS 213: Cognitive Psychology HS 224 Contemporary Short Fiction HS 226 Sociology of Cinema	AA 202N: Astronomical Techniques AA 204: Introduction to Space Exploration
5th : Minor 3	BSE 301: Introduction to Molecular Biology	CH 301: Functional Materials	HS 323: International Economics HS 321: History of Modern Indian Business	HS 311: Life and Thought of Gandhi HS 327: Mind, Action, and Technology HS 341: Appreciating Indian English Literature	AA 301: High Energy Astronomy AA 303: IoT for Space Applications
6TH : Minor 4			HS 325: Industrial Organization	HS 315: Sociology of Science and Technology HS 328 Philosophy and Film HS 330 Graphic Literature	

7th : (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)	(0-0-4-2) (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)	(0-0-4-2) (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)	(0-0-4-2) (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)	(0-0-4-2) (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)	(0-0-4-2) (minor project/field study/white paper/domain comprehension (Seminar)/Lab course)

Syllabi of
Computer Science and Engineering Courses

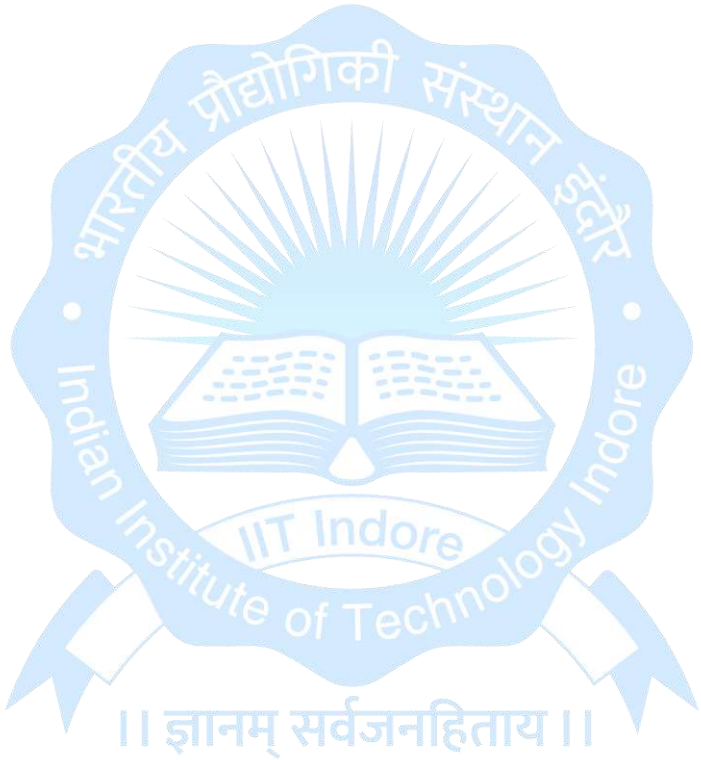
Course Code	**CS 103**
Title of the Course	**Computer Programming**
Credit Structure	L-T-P-Credits 2-0-0-2
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	None
Scope of the course	
Course Syllabus	This course provides an introduction to problem solving with computers using a modern language such as Java or C/C++. Topics covered will include: 1. Developer fundamentals such as editor, integrated programming environment, Turbo C++ and/or Microsoft Visual C++ Programming environment, modules, libraries. 2. Programming features: Machine representation, primitive types, arrays and records, objects, expressions, control statements, iteration, procedures, functions, and basic i/o. 3. Sample problems in engineering, science, text processing, and numerical methods.
Suggested Books	1. G. Dromey, **How to Solve It by Computer**, Prentice-Hall, Inc., Upper Saddle River, NJ, 1982 2. Coohoon and Davidson**, C++ Program Design: An introduction to Programming and Object- Oriented Design** (3rd edition), Tata McGraw Hill, New Delhi, 2003. 3. Yashwant Kanetkar, **Let us C**. Allied Publishers, 1998. 4. G. Polya, **How to Solve It** (2nd ed.), Doubleday and co. (1957). 5. **The Java Tutorial**, Sun Microsystems. Addison-Wesley, 1999.

Course code	**CS 201/MA 211**
Course category	**Department core**
Title of the course	Discrete Mathematical Structures
Credit Structure	L - T - P – Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Basic courses on mathematics
Objective(s)	This course will introduce the basic concepts of discrete mathematics and its applications.
Course Outcome	● Students will learn about discrete mathematical structures like sets, relations, functions, groups, graphs, etc. ● They will also learn about proof techniques and how to apply them to prove lemmas, theorems, etc.
Course Syllabus	● Elementary counting techniques ● Propositions and predicates, proofs and proof techniques. ● Sets, relations and functions, cardinality ● Posets and lattices: Dilworth`s theorem, inversion and distributive lattices ● Graph theory basics: paths, cycles, trees, connectivity ● Group theory: Lagrange`s theorem, homomorphisms, applications
Suggested Books	Textbooks: 1. K. H. Rosen, ***Discrete Mathematics and Its Applications***, Mc Graw Hill, 2019, ISBN: 9781259676512 Reference books: 2. R. P Grimaldi, ***Discrete and Combinatorial Mathematics***, Pearson, 2017, ISBN: 9788177584240

Course code	**CS 202 (From AY 2024-25 Onwards)**
Title of the course	**Automata Theory and Logic**
Course Category	Department core
Credit Structure	2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematical Structures
Objective(s)	To learn computation models, and classifying problem based on their solvability, and time taken by them on the computation models
Course Outcomes	Students will learn computation models and classification of problems based on how efficient they get solved on these models
Course Syllabus	● Finite state machines DFA/NFA/epsilon NFAs ● Regular expressions. Properties of regular languages. Pumping Lemma, Non-regularity, Myhill-Nerode Theorem. ● Push down automata. Properties of context-free grammar, Pumping Lemma for context-free grammar ● Turing hypothesis, Turing computability, Nondeterministic, multi tape and other versions of Turing machines, undecidability, The halting problem, post`s correspondence problem ● Complexity theory, P, NP, reducibility, NP-hard, NP-completeness
Suggested Books	Textbooks: 1. J.E. Hopcroft, R. Motwani, and J. D. Ullman, ***Introduction to Automata Theory, Languages and Computation***, Pearson Education, 2006. ISBN: 0-201-44124-1 2. H. R. Lewis, and C. H. Papadimitriou, ***Elements of the Theory of Computation***, Prentice Hall Inc, 1981. ISBN: 0-13-262478-8 Reference Books: 3. Michael Sipser, ***Introduction to the Theory of Computation***, Thomson, 2006. ISBN: 0-534-95097-3

Course code	**CS 203/ MA 213**
Title of the course	**Data Structures and Algorithms**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Computer Programming
Objective(s)	● This Course is designed to provide an introduction to the theory and practice of different data structures. ● This course will also provide familiarity with the algorithms for those data structures.
Course Outcomes	Students will learn the uses of data structures to make efficient algorithms
Course Syllabus	● Introduction to data structures, Abstract data types, Analysis of algorithms, Introduction to complexity analysis and measures. ● Arrays – operations and addressing, Linked list (singly, doubly, and circular), Stack ADT and its applications in expression evaluation and recursion, Queue ADT and its variants such as circular queues and double-ended queues. Hashing and hash tables, Recursion. ● Tree ADT, Binary trees – properties and traversals, Binary search trees, Height balanced trees -- AVL trees, Binary heaps, and priority queues. ● Graph ADT, Graph representation, Graph traversal – breadth-first search, depth-first search, and topological ordering, Connected components, cut-vertices, 2-connected components ● Algorithms and data structures for sorting and searching, Order statistics.
Suggested Books	Textbooks: 1. S. Sahni, ***Data structures, algorithms, and applications in C++***, McGraw-Hill, 1998, ISBN: 978-0929306322 2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, ***Introduction to Algorithms***, (3rd Edition), Prentice Hall, 2009. ISBN: 978-81-203-4007-7 Reference Books: 3. D. E. Knuth, ***The Art of Computer Programming: Fundamental Algorithms***, Vol. 1 (3rd Edition, 1997) and Vol 3, (2nd Edition, 1998),

	Addison-Wesley Professional. ISBN: 978-0137935109 4. M.T. Goodrich, R. Tamassia, and D. Mount, ***Data Structures and Algorithms in C++***, 2nd Edition, Wiley, 2011. ISBN: 978-0-470-38327-8



Course code	**CS 204**
Title of the Course	**Design and Analysis of Algorithms**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Discipline	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This is an introductory course in the field of computer algorithms.
Course Outcomes	At the end of the course, students will know the basics of ● algorithm analysis, ● algorithm design, and ● different problem classes.
Course Syllabus	● **Algorithm Analysis:** Time and Space Complexity; Computational Tractability (Best, Average and Worst Cases), Asymptotic Bounds (Lower, Upper and Tight Bounds). ● **Algorithm Design**: Divide and Conquer; Greedy, Dynamic Programming, Branch and Bound. ● **Problem Classes:** Reducibility and Intractability, P, NP, PSPACE, NP-Complete, and NP-Hard.
Suggested Books	Textbooks: 1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, ***Introduction to Algorithms*** (Eastern Economy Edition), 3rd Edition, PHI Learning Pvt. Ltd. (Originally MIT Press), 2010. ISBN: 978-8120340077 Reference books: 2. J. Kleinberg and E. Tardos, ***Algorithm Design***, 2nd Edition, Pearson Education, 2022. ISBN: 978-0132131087

Course Code	**CS 205** [from AY 2010-11 to AY 2013-14]
Title of the Course	**Abstractions and Paradigms for Programming**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	Review of the program development process, Issues in program design, Structured programming, Data and control abstractions, Programming with assertions. Reasoning about programs and proving correctness of programs. Ideas behind imperative, applicative, object oriented and logic programming paradigms such as typing, expressions, pure functions, recursion, higher order functions, encapsulation, inheritance, goal satisfaction, backtracking, unification. Some of the ideas behind the implementation of the paradigms.
Suggested Books	1. H. Abelson, G.J. Sussman, and J. Sussman, **Structure and Interpretation of Computer Programs** (2nd edition), The MIT Press, 1996. 2. D.A. Watt, **Programming Language Concepts and Paradigms**, Prentice-Hall, 1990. 3. R. Sangal, **Programming Paradigms in LISP**, McGraw Hill, 1991.

Course Code	**CS 206**
Title of the Course	**Logic Design**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Knowledge of Basic Electronics and Electrical Engineering
Scope of the course	
Course Syllabus	Switching theory: Introduction to number systems, Computer arithmetic, switching function and logic circuits, Combinational Logic, Canonical Logic Forms, K-maps Standard logic (SSI, MSI) vs. programmable logic (PLD, PGA). Finite state machine design: logic, minimization and races. Arithmetic unit, Control unit design, Logic design applications in computer systems, Introduction to computer-aided design Software, FPGA overview, Introduction to design automation and design through Higher level languages like VHDL.
Suggested Books	1. M. Zwonlinski, Digital Systems Design with VHDL, Pearson Education, 2003. 2. R.H. Katz and G. Borriello Contemporary Logic Design (2nd edition), Prentice Hall, 2004. 3. S.H. Unger, The Essence of Logic Circuits, Prentice Hall Inc. Englewood Cliffs, NJ, 1989 4. Foundations of Digital Logic Design, World Scientific Singapore, 1998.

Course code	**CS 207N**
Title of the course	**Database and Information Systems**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This course will introduce the fundamentals of Database Management Systems and practical solutions to create, manipulate, and optimize Databases.
Course outcome	Students will learn about ● The fundamentals of database management ● handling databases
Course Syllabus	● **Introduction to Databases:** Nature of Business Systems and Data Processing, Database Architectures, Schema, Data Models, XML. ● **ER Model:** Entity, Attribute, Relationship, ER Diagrams, UML, Class Diagrams. ● **Relational model and query languages:** Relational algebra and calculus, SQL. ● **Database design and normalization**: Integrities, Anomalies, Functional Dependencies, Normal Forms. ● **Transactions:** Introduction to transactions, Serializability, Recoverability, Concurrency control and recovery. ● **Physical Organization of Databases:** Indexing and Hashing, Single-level indexing, Multi-level indexing, B and B^+ Trees. ● **Query processing and optimization**
Suggested Books	Textbooks: 1. A. Silberschatz, H.F. Korth, and S. Sudarshan, ***Database System Concepts***, 7th Edition, McGraw Hill, 2019. ISBN: 9780078022159 2. R. Elmasri and S. Navathe, ***Fundamentals of Database Systems***, 7th Edition, Pearson, 2015. ISBN: 978-0133970777 Reference books: 3. R. Ramakrishnan and J. Gehrke, ***Database Management Systems***, 3rd Edition, McGraw Hill, 2002. ISBN: 978-0072465631 4. C. J. Date, ***Introduction to Database Systems***, 8th Edition, Pearson, 2003. ISBN 978-0321197849

Course Code	**CS 208**
Title of the Course	**Software Engineering**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	None
Scope of the course	
Course Syllabus	Software Situation: problems & causes; Role of Software Engineering; Software Development Paradigms. Function-oriented Methodology: System Engineering Overview; Function-Oriented Modeling Techniques; Function-Oriented Requirements Analysis; Correctness Criteria for Requirements Models; Reducing Complexity; Data Dictionary; Process Specification; Data Design; Architectural Design; Flow Analysis and Conversion Techniques; Design Refinement Measures; Procedural Design; User Interface Design. Object-oriented Methodology: Modeling of Software Requirements and Specifications with Use-Case Diagrams; Object-Oriented Modeling based on UML: Notations, Diagrams, Relationships, Modeling procedures & Applications; System Architecture; User-Interface Design; Game Interfaces and Web-based SE. Implementation: Procedural Design and Implementation. Stepwise Refinement. Software Project Management: Concerns of Management; Project Planning; Measurement and Metrics; Cost Estimation; Scheduling and Team Organization; Overview of SQA; SQA Techniques: qualitative and quantitative; Software Maintenance; Overview of Software Configuration Management; Software Configuration Items and Change Control. Advanced Topics: Component-based Software Engineering; Real-time Software Engineering; Clean-room Software Engineering
Suggested Books	1. R. S. Pressman, Software Engineering: A Practitioner's Approach (6th Edition), McGraw-Hill, 2006. 2. I. Sommerville, Software Engineering (5th Edition), Addison-Wesley, 1996. 3. C. Ghezzi, J. Mehdi. and M. Dino, Fundamentals of Software Engineering, Prentice-Hall, 1991.

Course code	**CS 209**
Title of the course	**Logic Design**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	None
Objective(s)	The scope of the course covers the Boolean functions, Boolean arithmetic, combinational circuits, sequential circuits and programmable logic devices.
Course Outcomes	Students will learn about ● Boolean arithmetic, combinational and sequential circuits ● programmable logic devices.

Course Syllabus	● **Number systems and codes:** Digital systems, Binary numbers, Number base conversions, Representation of Negative Numbers, Complements, Error detecting and correcting codes-hamming codes.
	● **Boolean algebra and logic gates:** Basic Definitions, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions.
	● **Gate-level minimization:** The Map Method - K-map 4 variable, Combinational Logic Circuits: Analysis Procedure, Binary Adder-Subtractor, Parallel Adder, Carry Look Ahead Adder.
	● **Sequential circuits:** Latches, Flip-Flops-SR, D, JK and T, realization of FFs, synchronous and asynchronous sequential circuits-State table, and state diagrams, State reduction, ASM Charts, Shift Registers-SISO, SIPO, PISO, PIPO, Design of counters-Modulo-n, Johnson, Ring, Up/Down, Design of Serial Adder, Serial Multiplier, FSM, Moore and Mealy machines -Sequence detector, PLDs.
	● **Practical components:** CAD Tool: Intel Quartus/Xilinx ISE
	  ● Design and implementation of 4-bit digital comparator (without using XOR/XNOR gates).
	  ● Design and implement at RTL BCD to 7-segment display converter.
	  ● Flip-Flops and its Applications.
	  ● Implement the RTL circuit and VHDL of the following sequential circuits: JK Flip Flop.
	  ● Implement the RTL circuit and VHDL of the following sequential circuits: T-FF.
	  ● Implement the RTL circuit and VHDL of the following sequential circuits: D-FF.
	  ● Designing Counters.
	  ● Design an Asynchronous Mod 10 counter using D- Flip Flop.

Suggested Books	Textbooks: 1. M. Morris Mano and Charles R. Kime, ***Logic and Computer Design Fundamentals***, 5th Edition, Prentice Hall, 2015. ISBN: 978-0133760637 2. R.H. Katz and G. Borriello, ***Contemporary Logic Design*** (2nd edition), Prentice Hall, 2004. ISBN: 978-0201308570

Course code	**CS 210**
Title of the course	**Computer Architecture**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Logic Design
Objective(s)	● The scope of the course covers computer organization and architectures (MIPS-32 bit). ● The students will learn the concepts of computer technology, instruction set design, computer arithmetic, data path and control unit design of processors and memory systems.
Course Outcomes	Students will learn about ● computer performance and processor architecture ● computer memory systems.
Course Syllabus	● **Introduction:** Computer Technology, performance trends, machine, power trends, memory capacity, # instructions, CPU time, Elapsed time, User CPU time, MIPS, Amdahl's law, problem solving based on performance parameters. ● **Instruction Set:** Instruction set design and architecture, general computer systems organization, computer instructions, addressing modes, ISA classes, MIPS processor arithmetic, instruction format, floating point numbers, double floating point representation, IEEE 754 floating point standard, FP to decimal, and decimal to FP. ● **Register Transfer and Datapath Structures:** Data path structure, Register transfer (RT), CPU pipelining basics, hazards: structural, control, data, preventing pipelining, data forwarding, stalling, etc. ● **Memory:** Memory hierarchy design, cache memory. Direct mapped cache Associative cache, Fully mapped cache, multi-level cache, tags, cache datapath, multi-word cache etc Virtual memory, TBL, physical disk, Multiprocessors, cache coherence, cache consistency.
Suggested Books	Textbooks: 1. D. Patterson, J. Hennessy, ***Computer Organization and Design***, 6th edition, 2020, Elsevier, ISBN: 9780128201091 2. M. M Morris, ***Computer System Architecture***, 3rd edition, Pearson, 2017. ISBN: 9789332585607

Course code	**CS 211**
Title of the course	**UX/UI Design**
Course category	Departmental Elective
Credit Structure	L - T - P – Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Knowledge of Programming
Objective(s)	Understand the fundamental concepts, techniques, practices, workflows, and tools associated with the practice of user experience design. Understand how to approach UI design in web and mobile experiences.
Course Outcome	Learn main concepts in user experience design and understand the importance of user-centered perspective on UX design. Learn to design user interface by applying interaction design guidelines. Get knowledge about usability testing and undertake further training in this area.
Course Syllabus	• **Introduction:** Introduction to User Experience (UX) design, Understanding users, Principles of UX design, Prototyping. • **Humans, Technology, and Design Basics:** Humans on the Web, Web use, Web design basics, Principles of UX design. **Designing for Human Psychology:** Human perception, Visual structure - Gestalt principles, Human vision, Memory, Attention. • **UX Design Process:** Defining problem, User research, Analysis of user insights, Designing and planning the user interface, Information architecture and interaction design, Prototyping, Usability testing. • **Guidelines and Standards:** Universal-design principles and heuristic guidelines, Jakob Nielsen's principles, Arnold Lund's expert ratings of usability maxims, Empathy map • **Practical Components:** ○ Basic HTML, FTP, and 'uploading' files to a server. ○ HTML elements and Stylesheets ○ Graphic production and Photoshop ○ Imagemaps, intermediate graphics production, Animated GIFs ○ Dynamic HTML ○ Project front page and graphics
Suggested Books	Textbooks: 1. S. Krug, ***Don't Make Me Think, Revisited, A Common Sense Approach to Web Usability***, Third Edition, 2014, New Riders, ISBN-

	10: 9780321965516 2. J. Johnson, ***Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines***, Third Edition, Morgan Kaufmann Publishers In, 2020, ISBN: 978-0124079144 Reference Books: 3. D. Norman, ***Design of Everyday Things: Revised and Expanded***, New York: Basic Books, Expanded Edition, 2014, ISBN: 978-0465055715 4. J. J. Garret, ***The Elements of User Experience: User-Centered Design for the Web***, New Riders, Second Edition, 2010, ISBN: 978-0321683687 5. K. Goodwin, ***Designing for the Digital Age: How to Create Human-Centered Products and Services***, Wiley, 2009. ISBN: 978-0470229101

Course code	**CS 212**
Title of the course	**Foundation of Algebraic Graph Theory**
Course category	Department elective
Credit Structure	L -T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Linear Algebra, Discrete Mathematics
Objective(s)	This course is about ● the analysis of graph properties using matrix theory ● bounds on some intractable graph problems.
Course outcomes	The students will learn analyzing graph properties using matrix theory and bounds on different graph problems
Course Syllabus	● Matrices associated with graphs, adjacency matrix, Laplacian matrix, distance matrix, Seidel Matrix, Spectral Theorem ● Finding number walks, connected components, Counting number of spanning trees, Matrix-Tree Theorem(s) ● Algebraic connectivity, regular graphs, random walks, expanders, Ramanujan Graphs ● Graph Isomorphism problem, graphs determined by the eigenvalues ● Strongly regular graphs, Friendship Theorem, Spectral bounds on NP-hard problems on graphs
Suggested Books	Textbooks: 1. R. B. Bapat, ***Graphs and Matrices***, Hindustan Book Agency, 2014, ISBN: 978-1-4471-6568-2 2. C. Godsil and G. Royle, ***Algebraic Graph Theory***, Springer, 2001, ISBN: 978-0-387-95241-3 Reference books: 3. A. E. Brouwer and W. H. Haemers, ***Spectra of graphs***, Springer, 2011, ISBN: 978-1-4614-1938-9

Course code	**CS 213**
Title of the course	**Matrix factorizations and applications**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Basic linear algebra
Objective(s) (Objectives)	● This Course is designed to provide an introduction to matrix factorizations. ● This course will also provide familiarity with some algorithms related to matrix factorization.
Course Outcomes	Students will learn various matrix factorizations and their applications
Course Content	● Review of vector spaces, bases ● Cayley-Hamilton Theorem, triangulation, diagonalization, LU, LUP, decompositions ● Linear transformations, rotations, reflections, Gram-Schmidt orthogonalization, QR like decompositions, linear least-square problems ● Eigenvalues, Eigenvectors, normal matrices, eigenvalue decomposition, similar matrices ● Spectral theorem for Hermitian matrices, Cauchy-interlace theorem, Singular value decomposition, Moore-Penrose pseudoinverse
Suggested Books	Textbooks: 1. G. H. Golub, Charles F. Van Loan, ***Matrix Computations***, The Johns Hopkins University Press, 2013. ISBN: 978-1421407944. 2. D. S. Watkins, ***Fundamentals of Matrix Computations***, Wiley, 2010. ISBN: 978-0-470-52833-4. Reference Books: 3. G. Strang, ***Linear Algebra and Its Applications***, 4th Edition, Academic Press, 2006. ISBN- 978-8131501726.

Course code	**CS 214**
Title of the course	**Foundations of Hardware Security**
Course category	Department elective (Semester 4)
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Course Instructors	Prof. Anirban Sengupta and Dr. Bodhisatwa Majumdar
Pre-requisite, if any	Knowledge of Logic Design, Algorithms
Objective(s)	This course will introduce the basic foundations of Hardware Security and the obfuscation techniques about hardware DSP cores, different techniques for IP protection, hardware design of cryptographic primitives, side channel analysis, hardware Trojans, and physically unclonable functions.
Outcome of the course	• The students will learn the VLSI aspects that relate to hardware security. • The students will acquire the knowledge of different security threats that thwart hardware trust in the VLSI design cycle along with the countermeasures against such threats.
Course Syllabus	• **Introduction to Hardware (IP) Security: Forensic Detective Control and Obfuscation of DSP cores:** Hardware Security, Hardware Security of DSP Core, Security – Energy Tradeoff in High Level Synthesis for Hardware Security of DSP cores • **Forensic Detective Control using Hardware Steganography for IP Core Protection:** Threat Model, Selected Contemporary Approaches - Steganography vs Watermarking vs Cryptography, Limitations of Hardware Watermarking • **Forensic Detective Control using Digital Signature based Watermark for IP Core Protection:** Threat Models of an IP Core, Selected Contemporary Approaches • **Protection of Fault Secured IP Core using Digital Signature based Watermark:** Background on different IP Core Threats and Countermeasures, Threat Model, Selected Contemporary Approaches • **Multi-Level Watermark for IP Protection:** Discussion on Selected Approaches, Salient Features and Advantages of Multi-level Watermark, Embedding Signature as Secret-mark • **Hardware Design of Cryptographic Algorithms:** Hardware Design of Advanced Encryption Standard (AES), Efficient Design of Finite Field Arithmetic on FPGAs • **Side Channel Analysis (SCA):** Introduction to SCA techniques, Power-based SCA • **Hardware Trojans and PUFs:** Overview of Hardware Trojans, Their

	Effect on Circuit Reliability, Techniques of Hardware Trojan Insertion, PUFs: Root-of-trust for Hardware Security
Suggested Books	Textbooks: 1. A. Sengupta, ***Frontiers in Securing Hardware IP: Forensic Detective Control and Obfuscation***, IET, 2020. ISBN: 978-1839530319 2. D. Mukhopadhyay and R.S. Chakraborty, ***Hardware Security: Design, Threats, and Safeguards***, CRC Press, 2014. ISBN: 9780429066900 Reference books:

Course code	**CS 215**
Title of the course	**Mathematics for AI and ML**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Basic linear algebra and calculus
Objective(s)	This course is designed to provide an introduction to mathematical foundations, concepts, and constructs for artificial intelligence and machine learning algorithm design.
Course Outcomes	Students will develop a foundation such that advanced courses in this area could be taken (Artificial Intelligence, Machine Learning, Soft Computing, and Computational Intelligence).
Course Content	• **Linear Algebra and Matrix Analysis:** Systems of Linear Equations, Vector Spaces, Linear Independence, Basis and Rank, Orthonormal Basis, Orthogonal Complement, Orthogonal Projections, Rotations, Eigenvalue Decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation. • **Vector Calculus and Continuous Optimization:** Gradients of Vector-Valued Functions, Gradients of Matrices, Automatic Differentiation, Higher-Order Derivatives, Linearization and Multivariate Taylor Series, Unconstrained Optimization, Constrained Optimization and Lagrange Multipliers. • **Probability and Distributions:** Probability Space, Discrete and Continuous Probabilities, Sum Rule, Product Rule, and Bayes' Theorem, Summary Statistics and Independence, Gaussian Distribution, Conjugacy and the Exponential Family, Change of Variables/Inverse Transform. • **Models and Data:** Models Learning and Selection, Empirical Risk Minimization, Parameter Estimation, Probabilistic Modeling and Inference, Directed Graphical Models, Bayesian Linear Regression, Dimensionality Reduction with Principal Component Analysis, Density Estimation with Gaussian Mixture Models, Classification with Support Vector Machines

Suggested Books	Textbooks: 1. M. P. Deisenroth, A. A. Faisal, and C. S. Ong., ***Mathematics of Machine Learning***, Cambridge University Press, 2020. ISBN: 978-1-1084-5514-5. 2. A. Antoniou and W.-S. Liu, ***Practical Optimization: Algorithms and Engineering Applications***, Springer, 2007. ISBN: 978-0-3877-1106-5. Reference Books: 3. C. Meyers, ***Matrix Analysis and Applied Linear Algebra***, SIAM, 2023. ISBN: 978-1-6119-7745-5. 4. J. K. Blitzstein and J. Hwang, ***Introduction to Probability*****,** Chapman and Hall/CRC Texts in Statistical Science, 2019. ISBN: 978-1-1383-6991-7 5. T. Hastie, R. Tibshirani, and J. Friedman, ***The Elements of Statistical Learning: Data Mining, Inference, and Prediction*****,** Springer Series in Statistics, 2016**.** ISBN: 978-0-3878-4857-0.

Course Code	**CS 216**
Title of the Course	**Introduction to Blockchain**
Course Category	Departmental Elective/ Institute Elective
Credit Structure	L-T-P–Credits 2-1-0-3 (½ semester)
Name of the Concerned Department	Computer Science and Engineering
Prerequisite if any	Programming and Data Structure
Course Objective	This course introduces students to blockchain, and develops familiarity of current technologies, tools, and implementation strategies.
Course Outcomes	Students will develop a clear understanding of Blockchain and its applications.
Course Content	• Primers on Blockchain - Motivation for Blockchain Systems. Introduction to Peer-to-Peer systems and Distributed systems • Preliminaries and Mathematical Background - Cryptographic primitives like hashing, digital signature, encryption, Concepts of basic probability like sample space, independent event, Bayes' theorem, expectation • Introduction to Bitcoin: Bitcoin - Wallet - Blocks - Merkle Tree - hardness of mining - transaction verifiability - anonymity, Proof-of-Work. • Alternate Consensus Mechanisms - Proof of Stake, Proof of Space, Proof of Space time, Proof of Burn, Proof of Authority
Suggested Books	Textbooks: 1. A. M. Antonopoulos, ***Mastering Bitcoin: Programming The Open Blockchain***, O'Reilly, 2017. ISBN: 9789352135745 2. A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, ***Bitcoin and cryptocurrency technologies: a comprehensive introduction***, Princeton University Press, 2016. ISBN: 9780691171692 Reference books: 3. A. M. Antonopoulos and G. Wood, ***Mastering Ethereum: Building Smart Contracts and Dapps***, O'reilly Media, 2018. ISBN: 978-9352137961

Course Code	**CS 218**
Title of the Course	**Programmable and Interoperable Blockchains**
Course Instructors	Dr. Subhra Mazumdar and Dr. Bodhisatwa Mazumdar
Course Category	Departmental Elective
Credit Structure	L-T-P–Credits 2-1-0-3 (½ semester)
Name of the Concerned Department	Computer Science and Engineering
Prerequisite if any	Programming and Data Structure, Introduction to Blockchain
Course Objective	This course introduces students to blockchain, and develops familiarity of current technologies, tools, and implementation strategies.
Course Outcomes	Students will develop a clear understanding of Blockchain and its applications.

Course Content	• Introduction to Ethereum:
	  ○ Ethereum and Smart Contracts, The Turing Completeness of Smart Contract Languages and verification challenges, Solidity Programming
	  ○ Ethereum Virtual Machine (EVM) - Wallets for Ethereum
	  ○ Consensus in Ethereum: Proof of Stake, Casper: Friendly Finality Gadget,Gasper, Rewards and Penalties
	• Comparison of Proof of Work vs Proof of Stake:
	  ○ Identifying the pros and cons of each consensus mechanism, issue of Scalability and Security
	  ○ Availability-Finality Dilemma and its Resolution
	  ○ Some mechanisms to study: Ebb-and-Flow, Snap-and-Chat
	• Decentralized Identity Management:
	  ○ Risk of data breaches in centralized identity management systems
	  ○ Decentralized Identity Foundations
	  ○ Self Sovereign Identity and Decentralized Identifiers, Construction
	• Blockchain Interoperability:
	  ○ Public vs permissioned blockchain
	  ○ Asset and Data Interoperability
	  ○ Hash-Timed Lock Contract (HTLC), Sidechains, Bridge-based interoperability

Suggested Books	Textbooks: 1. A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, ***Bitcoin and cryptocurrency technologies: a comprehensive introduction***, Princeton University Press, 2016. ISBN: 9780691171692 2. A. M. Antonopoulos, ***Mastering Bitcoin: Programming The Open Blockchain***, O'Reilly, 2017. ISBN: 9789352135745 Reference books: 3. A. M. Antonopoulos and G. Wood, ***Mastering Ethereum: Building Smart Contracts and Dapps***, O'reilly Media, 2018. ISBN: 978-9352137961

Course code	**CS 302N**
Title of the course	**Computer Graphics and Visualization**
Course Category	Department core
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Programming
Objective(s)	Introduce the theory and practice of computer graphics and an insight of modern graphics systems.
Course Outcomes	● Understand the basic principles of computer graphics primitives. ● Design application-specific computer graphics programs.
Course Syllabus	● **Introduction:** Introduction to computer graphics, Graphics hardware and display devices, Raster and vector graphics, Pixel concept, Raster scan algorithms. ● **2D Computer Graphics:** Homogeneous coordinates, Window and view port, 2D viewing pipeline, 2D geometric transformation. **3D Computer Graphics:** 3D viewing pipeline, 3D geometric transformations, Planner projections. ● **Clipping:** 2D and 3D line and polygon clipping algorithms. ● **Visible Surface Detection**: Planner surface representation, Visible surface determination algorithms. ● **Light, Shading and Color Models:** Illumination and shading models, RGB, CMYK and YCbCr colors model ● **Solid Model:** Solid representation, Regularized Boolean set representation, Sweep and primitive representation, CSG, Quad tree, Octree, BSP ● **Practical Components:** ○ Basics of graphics libraries like OpenGL/DirectX/Others ○ Visualizing graphics algorithms using graphics libraries ○ Visualizing 3D scenes using graphics libraries. ○ Small game designing projects and scientific visualization with graphics libraries
Suggested Books	Textbooks: 1. J. F. Hughes, A. V. Dam, M. McGuire, D. F Sklar, J. D. Foley, S. K. Feiner, and K. Akeley, ***Computer Graphics: Principles and Practice***, Pearson, 2018, ISBN: 978-0321399526. 2. D. D. Hearn, M. P. Baker, and W. Carithers., ***Computer Graphics with***

	Open GL, Pearson, 2015. ISBN: 9780136053583. Reference Books: 1. M. K. Pakhira, ***Computer Graphics, Multimedia and Animation***, PHI, 2010, ISBN: 9788120341272. 2. D. D Hearn and M. P. Baker, ***Computer Graphics, C Version***, Pearson, 2002, ISBN: 9788177587654. 3. F. S. Hill, Jr. and S. Kelley, ***Computer Graphics Using OpenGL***, Pearson, 2007, ISBN: 978-0131496705. 4. F. Luna, ***Introduction to 3D Game Programming with DirectX 12***, First Eds., Mercury Learning and Information, 2016. ISBN: 9781942270065.

Course code	**CS 253**
Title of the course	**Data Structures and Algorithms Lab**
Course Category	Department core
Credit Structure	L - T - P - Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Computer Programming
Objective(s)	This Course is designed to provide ● an introduction to the theory and practice of different data structures ● familiarity with the algorithms for those data structures
Course Outcomes	Students will learn uses of data structures to make efficient algorithms.
Course Syllabus	● Implementation of array, linked list, stack, and queue ● Implementation of tree and graph data structure ● Implementation of sorting and searching, ● Implementation of Hash and hash tables and order statistics.
Suggested Books	Textbooks: 1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, ***Introduction to Algorithms***, (3rd Edition), Prentice Hall, 2009. ISBN: 978-81-203-4007-7 Reference Books: 2. D. E. Knuth, ***The Art of Computer Programming: Fundamental Algorithms***, Vol. 1 (3rd Edition, 1997) and Vol 3, (2nd Edition, 1998), Addison-Wesley Professional. ISBN: 978-0137935109 3. M.T. Goodrich, R. Tamassia, and D. Mount, ***Data Structures and Algorithms in C++***, 2nd Edition, Wiley. 2011. ISBN: 978-0-470-38327-8

Course code	**CS 254/ MA 254**
Title of the Course	**Design and Analysis of Algorithms Laboratory**
Course Category	Department core
Credit Structure	L - T - P - Credits: 0-0-3-1.5
Name of the Concerned Discipline	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This is an introductory course in the field of computer algorithms.
Course Outcomes	At the end of the course, students will know the basics of ● algorithm analysis and design ● different problem classes.
Course Syllabus	● Runtime analysis of different sorting algorithms and linked lists in best-case, worst-case, and average-case. ● Implementation and analysis of algorithms based upon the following design techniques: ○ Divide and Conquer Strategy (Closest Pair of Points, Integer Multiplication, Matrix Multiplication, Fast Fourier Transform etc.). ○ Greedy Strategy (Interval Partitioning, Dijkstra's Algorithm, Minimum Spanning Tree etc.). ● Dynamic Programming Strategy (Weighted Interval Scheduling, Sequence Alignment, Bellman-Ford Algorithm etc.).
Suggested Books	Textbooks: 1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, ***Introduction to Algorithms*** (Eastern Economy Edition), 3rd Edition, PHI Learning Pvt. Ltd. (Originally MIT Press), 2010. ISBN: 978-8120340077 Reference books: 2. J. Kleinberg and E. Tardos, ***Algorithm Design***, 2nd Edition, Pearson Education, 2022. ISBN: 978-0132131087

Course Code	CS 255
Title of the Course	Abstractions and Paradigms for Programming Lab
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	This lab course is to be centered around problems and applications that demonstrate the main themes of the associated theory course CS 205. This laboratory would include the sessions for the following topics: 1. Functional Programming Basics using Scheme: Expressions, Naming, Combinations, Procedures, Conditions. 2. Recursion: Procedure v/s Process; Recursive v/s Iterative 3. Scheme: Higher-Order procedures, let, lambda; Procedures as Arguments, General Methods. 4. Lists: Basic Operations using Lists in Scheme 5. Matrix Manipulation in Scheme 6. Tags & Multiple Representations in Scheme 7. Object-Oriented Programming: Classes, Objects using Java 8. Inheritance, Polymorphism, Message Passing in Java 9. Concurrent Programming: Creating Thread, Use Different Functions Related Thread in Java 10. Thread Synchronization & Producer Consumer Problems in Java 11. Logic Programming using Prolog: Domain Variables, Specification of Constraints, Solution Space. 12. Imperative Programs, Loop Invariants.
Suggested Books	Same as CS 205

1.	Course Code	CS 256
2.	Title of the Course	Logic Design Lab
3.	Credit Structure	L-T-P-Credits 0 -0-3-1.5
4.	Name of the Concerned Department	Computer Science and Engineering
5.	Pre–requisite, if any	Knowledge of Electronics and Electrical Engineering Lab
6.	Scope of the course	
7.	Course Syllabus	Experiments with Logic Building Blocks using SSI/MSI, Experiments on Design and/or use Minimization tools. Use of VHDL and simulation in Logic Design. A small project on design with the use of tools and MSI and/or PLDs. FPGA basics and programming.
8.	Suggested Books	Same as the associated theory course CS 206: Logic Design

Course code	CS 257
Title of the course	Database and Information Systems Lab
Course Category	Department core
Credit Structure	L - T - P - Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This course will introduce the fundamentals of Database Management Systems and practical solutions to create, manipulate, and optimize Databases.
Course outcome	Students will learn and implement about ● The fundamentals of database management ● handling database
Course Syllabus	● Use of database systems supporting interactive SQL. ● Two-tier client-server applications using JDBC or ODBC. Three-tier web applications using Java servlets/JDBC or equivalent. ● Design of applications and user interfaces using these systems. ● Data analysis tools. ● Laboratory project.
Suggested Books	Textbooks: 1. A. Silberschatz, H.F. Korth, and S. Sudarshan, ***Database System Concepts***, 7th Edition, McGraw Hill, 2019. ISBN: 9780078022159 2. R. Elmasri and S. Navathe, ***Fundamentals of Database Systems***, 7th Edition, Pearson, 2015. ISBN 978-0133970777 Reference books: 3. R. Ramakrishnan and J. Gehrke, ***Database Management Systems***, 3rd Edition, McGraw Hill, 2002. ISBN: 978-0072465631 4. C. J. Date, ***Introduction to Database Systems***, 8th Edition, Pearson, 2003. ISBN 978-0321197849

Course Code	CS 258
Title of Course	Software Engineering Laboratory
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Department	Computer Science and Engineering
Pre-requisite, if any	Should be enrolled in parallel in CS 208 or should have already taken and successfully completed the CS 208 course
Scope of the course	To provide students with an environment in which to experience the process of Software Development by working through 'real world' projects
Course Structure	Students would be made to go through and experience the various phases of the Software Development Life Cycle by working on a real project and sequentially working through the phases. The Software Developments Phases include broadly: 1) Requirements Elicitation 2) Software Design 3) Software Development 4) Software Testing 5) Software Maintenance
Suggested books	1.R. S. Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill, 1982 2. I. Sommerville, Software Engineering, Addison-Wesley, 1996

Course Code	CS 261 [for AY 2010-11 only]
Title of the Course	Program Development and Software Design Lab - I
Credit Structure	L-T-P-Credits 0-1-4-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Knowledge of Computer Programming
Course Syllabus	Longer Programs based on creating and manipulating various data structures. The lab work includes documentation as well as testing.
Scope of the course	
Suggested Books	1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, **Introduction to Algorithms**, (2nd Edition), Prentice Hall India, 2002. 2. D. E. Knuth, **The Art of Computer Programming**, Vol. 1 and 3, (2nd Edition), Addison-Wesley, 1998.

Course Code	CS 262 [for AY 2010-11 only]
Title of the Course	Program Development and Software Design Lab - II
Credit Structure	L-T-P-Credits 0-1-4-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Knowledge of Computer Programming
Scope of the course	
Course Syllabus	Programs based on principles of software design and involving various data structures. The lab work includes documentation as well as testing.
Suggested Books	1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, **Introduction to Algorithms**, (2nd Edition), Prentice Hall India, 2002. 2. D. E. Knuth, **The Art of Computer Programming**, Vol. 1 and 3, (2nd Edition), Addison-Wesley, 1998.

Course code	CS 264
Title of the course	Computer Architecture Lab
Course Category	Department core
Credit Structure	L - T - P - Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Logic Design
Objective(s)	● The scope of the course covers computer organization and architecture (MIPS-32 bit). ● The students will learn the concepts of computer technology, instruction set design, computer arithmetic, data path and control unit design of processors and memory systems.
Course Outcomes	Students will learn about ● computer performance and processor architecture ● computer memory systems.
Course Syllabus	● Comparison of various modern processor architectures. ● Case studies of various performance issues. ● Implementation and handling of strings in MIPS assembly language ● Implementation and handling of loops in MIPS assembly language. ● Implementation and handling of while condition in MIPS assembly language. ● Implementation of functions in MIPS. ● Implementation of conditional branching in MIPS. ● Implementation of consecutive integers in MIPS. ● Implementation of switch case in MIPS. 10: Implementation of dual integer function in same MIPS code. ● Implementation of factorial function in MIPS. ● Implementation of strcpy in MIPS. ● Basic programming in Quartus II tool. ● Coding of CPU Register set design (VHDL, Simulation). ● Design a 32-bit ALU in VHDL and simulate the code.
Suggested Books	Textbooks: 1. D. Patterson, J. Hennessy, ***Computer Organization and Design***, 6th edition, 2020, Elsevier, ISBN: 9780128201091 2. M. M Morris, ***Computer System Architecture***, 3rd edition, Pearson, 2017. ISBN: 9789332585607

Course Code	CS 302
Title of the Course	Computer Graphics and Visualization
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Computer Programming, Data Structures and Algorithms
Scope of the course	This Course is designed to provide an introduction to the theory and practice of computer graphics and an insight of modern computer graphics systems. Students will understand the basic principles of computer graphics primitives and able to design application specific computer graphics program. This course will also provide familiarity with key algorithms for modelling and rendering graphical data.
Course Syllabus	**Introduction:** Basic of Computer Graphics. **Graphics Systems and Models:** Raster System; Vector System; Scan Conversion; 2D and 3D Graphics Model **Graphics Hardware:** Display Devices; Input Devices; Hard Copy Technology, Display Processors **Raster Graphics Algorithm:** Pixel Concept; Line, Circle, Ellipse, Polygon drawing Algorithms **Visualization Algorithm for Raster Graphics:** Colouring, Filling Scan Conversion Algorithms **2D Computer Graphics:** Homogeneous Coordinates; Window and View Port; 2D Geometric Transformation; 2D Viewing Pipeline **3D Computer Graphics:** Planner Projections; Vanishing Points; 3D Viewing Pipeliine, 3D Geometric Transformations **Colour, Light and Shading:** RGB Colour Model, CMYK Colour Model; YCbCr Color Model, Light Sources; Achromatic and Coloured Light; Illumination and Shading Model; Shadow **Curve and Surface Representation:** Polygon Meshes; Cubic Curves; Bicubic Surfaces. **Solid Model:** Solid Representation; Regularized Boolean Set Representation; Sweep and Primitive Representation, B-Reps; CSG; Quad Tree; Octree; BSP **Clipping:** 2D and 3D Line and Polygon Clipping Algorithms **Visible Surface Detection**: Planner Surface Representation; Visible Line Determination; List Priority Algorithm; Area Subdivision Algorithm; Z-Buffer Algorithm; Visible Surface Detection for BSP and Octree Representation; Ray Tracing **Visualization/Rendering:** Physical Description of Rendering, Image-order and Object-order; Surface and Volume Rendering; Transparency and Alpha Values; Realism; Aliasing and Anti-Aliasing; 3D Texture Mapping;

	Visualization Pipeline: Data Acquisition; Data Reduction; Visibility Transformation; Viewing Transformation and Rendering
Suggested Books	1. D. Hearn, M. P. Baker, ***Computer Graphics. C Version***, Pearson Education , 2nd Eds, 1997 2. D. Hearn, M. P. Baker, ***Computer Graphics with OpenGL***, Pearson Education India , 3rd Eds, 2004 3. F.S. Hill. ***Computer Graphics Using Open GL.*** Prentice Hall. 2001 4. John F. Hughes, Andries van Dam, James D. Foley, Morgan McGuire, Steven K. Feiner, David F. Sklar, Kurt Akeley, ***Computer Graphics, Principles and Practice***, Addison Wesley, 3rd Eds, 2014. 5. W. Schroeder, K. Martin, and B. Lorensen, ***The Visualization Toolkit***, (2nd Edition), Prentice-Hall, Inc., 1998. 6. M. K. Pakhira, ***Computer Graphics, Multimedia and Animation***, PHI, 2nd Eds, 2010

Course code	CS 303/ MA 303
Title of the course	Operating Systems
Course Category	Department core
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	NA
Objective(s)	This course will introduce the basic components of operating systems and functionalities.
Course outcome	Understanding basic functionalities of operating system for efficient performance of the processes
Course Syllabus	● **Introduction:** Overview of important features of computer architectures for OS operation; Service and system performance ● **Multiprogramming:** Concurrency and parallelism; Processes and threads; Process synchronization; Process deadlocks ● **Memory management:** Paging; Segmentation; Virtual memory ● **File systems:** File operations. File protection ● **Case Studies:** Case studies of contemporary operating systems
Suggested Books	Textbooks: 1. A. Silberschatz, P.B. Galvin, and G. Gagne, ***Operating System Principles***, 7th edition, John Wiley, 2005. ISBN 9788126509621 2. A. Silberschatz, P.B. Galvin, and G. Gagne, ***Operating System Concepts***, 9th edition, Wiley, 2018. ISBN 978-1-118-06333-0 Reference books: 3. W. Stallings, ***Operating Systems: Internals and Design Principles***, 5th edition, Pearson Education, 2005. ISBN 978-0-13-467095-9

Course Code	**CS 304** [from AY 2010-11 to 2014-15]
Title of the Course	**Artificial Intelligence**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Data Structures and Algorithms
Scope of the course	
Course Syllabus	**Basics of problem-solving:** problem representation paradigms, state space, satisfiability vs optimality, pattern classification problems, example domains. **Search Techniques:** Problem size, complexity, approximation and search; depth, breadth and best search; knowledge based problem solving, artificial neural networks. **Knowledge representation:** First order and non-monotonic logic; rule based, frame and semantic network approaches. **Knowledge Acquisition:** Learnability theory, approaches to learning. **Uncertainty Treatment:** formal and empirical approaches including Bayesian theory, belief functions, certainty factors, and fuzzy sets. **Detailed Discussion from Example Domains:** Industry, Language, Medicine, Verification, Vision, Knowledge Based Systems. **Languages and Machines:** AI languages and systems, special purpose architectures.
Suggested Books	1. S. Russell and P. Norvig, **Artificial Intelligence: A Modern Approach**, Prentice Hall Series in AI, 1995. 2. M. Stefik, **Introduction to Knowledge Systems**, Morgan Kaufman, 1995. 3. P.H. Winston, **Artificial Intelligence** (3rd edition), Addison Wesley, 1995. 4. E. Rich and K. Knight, **Artificial Intelligence**, Tata McGraw Hill, New Delhi 1992. 5. E. Charniack and D. McDermott, **Artificial Intelligence**, Addison Wesley, 1987. 6. 7. N.J. Nilsson, **Principles of Artificial Intelligence**, Morgan Kaufman, 1985.

Course code	CS 304N
Title of the course	Computational Intelligence
Course Category	Department core
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Computer Programming, Data structure, and Design and Analysis of Algorithm
Objective(s)	Basics of machine learning techniques
Course Outcomes	Understanding of machine learning techniques and implementation
Course Syllabus	● **Introduction:** Overview, Basics of Problem solving as an Artificial Intelligence problem, Computational Intelligence, Applications. Intelligent Search techniques, Knowledge representation, ● **Methodologies:** Computational intelligence methodologies; Learning, adaptation: Artificial neural networks: feed-forward, recurrent and multi-layer architectures; Supervised and unsupervised learning; Characteristics: adaptability, fault tolerance, generalization; limitations of neuro-computing. ● **Different learning algorithms:** Perceptron, Back propagation, Hopefield, Kohenen networks. Uncertainty treatment: Fuzzy sets - Basic Definition; Fuzzy-set- theoretic Operations – Member Function Formulation and Parameterization – Fuzzy Rules and Fuzzy Reasoning, Fuzzy If-Then Rules Hybrid computational learning : Fuzzy Neural Networks and Evolutionary Algorithms ● **Detailed Discussion from Example Domains:** Industry, Language, Medicine, Verification, Vision, Knowledge Based Systems etc.
Suggested Books	Textbooks: 1. S. Russell and P. Norvig, ***Artificial Intelligence: A Modern Approach***, Pearson, 2010. ISBN: 978-0136042594 2. E. Rich and K. Knight, ***Artificial Intelligence***, McGraw Hill Education, 2017. ISBN: 978-0070087705 Reference books: 3. J.S.R.J ang, C.T. Sun and E. Mizutani, ***Neuro-Fuzzy and Soft Computing***, Prentice Hall of India and Pearson Education, 2004. ISBN: 978-9332549883 4. D.E. Goldberg, ***Genetic Algorithms: Search, Optimization and Machine Learning***, Addison Wesley, 1989. ISBN: 9781584883883 5. S. Rajasekaran and G.A.V. Pai, ***Neural Networks, Fuzzy Logic and Genetic Algorithms***, Prentice Hall, 2003. ISBN: 9788120321861 6. R. Eberhart, P. Simpson and R. Dobbins, ***Computational Intelligence - PC Tools***, AP Professional, 1996. ISBN: 978-0122286308

Course Code	**CS 305**
Title of the Course	**Computer Architecture**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Logic Design
Scope of the course	
Course Syllabus	**Assembly Level Organization:** instruction formats, addressing mechanisms, Architecture and programming of 8085 and or x86 architectures, microprogramming, Arithmetic and Logic Unit. **Memory Systems:** memory hierarchy, main memories, cache, virtual memory, Pipeline processing. **Interfacing and Communication:** I/O, interrupts, buses. Multiprocessor and alternative architectures, Contemporary architectures Computer organization and architecture Lab Machine/Assembly programming, Design of basic computing units.
Suggested Books	1. J.L. Hennessey, D.A. Patterson, **Computer Architecture**: A Quantitative Approach (4th Edition), Morgan Kauffman, 2006. 2. W. Stallings, **Computer Organization and Architecture** (7th edition), Prentice Hall Inc., 2006 3. J.P. Hayes, **Computer Architecture and Organization** (3rd edition), McGraw-Hill Inc. 2002

Course Code	**CS 306**
Title of the Course	**Computer Networks**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Data Structures and Algorithms
Scope of the course	
Course Syllabus	Design of Computer Networking protocols at all layers: transmission media, data link protocols, media access control, routing and congestion control, admission control, traffic shaping and policing, Internet working (IP) and transport layer protocols (TCP). Performance analysis of networks.
Suggested Books	1. W. Stallings, **Data and Computer Communications** (6th edition), Prentice Hall, 2000. 2. S. Tannenbaum, **Computer Networks** (4th edition), Prentice Hall Inc., 2003. 3. F. Halsall, **Data Communications: Computer Networks and Open Systems** (4th edition), Addison-Wesley, 1996. 4. Walrand and Varaiya, **High Performance Communication Networks**, Morgan Kaufman, 1996. 5. D. E. Comer, **Internet working with TCP/IP: Principles, Protocols, Architecture** (3rd edition), Prentice Hall, 2000. 6. W. R. Stevens, **TCP/IP Illustrated** (Vol. I), Addison Wesley, 1994.

Course code	CS 307/ MA 307
Title of the Course	Optimization Algorithms and Techniques
Course Category	Department core
Credit Structure	L-T-P-Credits 2–1–0-3
Name of the Concerned Discipline	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This is an introductory course in the field of mathematical optimization.
Course Outcomes	At the end of the course, students will know ● The Basics of Optimization, ● Unconstrained and Constrained Optimization, and ● Linear and Quadratic Programming.
Course Syllabus	● **Introduction to Optimization and Math Foundation:** Type of Problems, Examples, Formulations, Applications, Notations, and Convexity. ● **Unconstrained Optimization**: Necessary and Sufficient conditions for a Minima; Linear Search and Trust Region Methods; Multi-dimensional Minimization - Steepest descent, Newton, Gauss Newton, Quasi Newton; One-Dimensional minimization - Dichotomous, Quadratic and Cubic Interpolation. ● **Constrained Optimization:** Conversion to Unconstrained, Lagrange Multipliers, Necessary and Sufficient Conditions for Minima (KKT), and Duality. ● **Linear Programming**: Necessary and Sufficient Conditions for a Minima for a Linear Program, Derivation and Implementation of Simplex, Starting Simplex, and Interior-Point Methods.
Suggested Books	Textbooks: 1. J. Nocedal and S. J. Wright, ***Numerical Optimization***, 1st Edition, Springer, 2006. ISBN: 78-1-4939-3711-0 Reference books: 2. A. Antoniou and W.-S.g Lu, ***Practical Optimization: Algorithms and Engineering Applications***, 2nd Edition, Springer, 2021. ISBN: 978-1-0716-0843-2

Course Code	CS 308
Title of the Course	Compiler Techniques
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Automata Theory and Logic, Data Structures and Algorithms, Abstraction and Paradigms in Programming
Scope of the course	
Course Syllabus	**Introduction:** Major compilation processes; Compiler phases; front end and back end partitioning. **Lexical Analysis:** Tasks and roles of lexical analyser; Regular expressions; Deterministic finite automata; *LEX* – a lexical analyzer generator. **Context-Free Grammars:** Formal grammar and Backus Naur Form; Derivations; Ambiguous, unambiguous and recursive grammars; Chomsky hierarchy; Parse trees and parsing concepts. **Syntax Analysis:** Top down parsing – recursive descent and LL(1) predictive parsers; First and Follow sets; LL(1) parse table construction; Bottom up and shift reduce parsing; LR parsing; Parse table constructions – *LR(0), SLR(1) and LALR(1); YACC* – a syntax analyser generator. **Extending the Parser:** Syntax directed approach; *YACC's* support for attribute evaluation; Inherited and synthesized attributes; symbol table; Type concepts; Syntax-directed semantic analysis; Run-time storage organization; Intermediate languages – three address code; Syntax-directed intermediate code generation. **Introducing Compiler Backend:** Code optimization techniques and concepts; Target code generation. **A Complete Compiler:** The grammar specification; scanner; parser; code generation; Building and running the compiler; The Assembler and the virtual machine.
Suggested Books	1. A.V. Aho, M.S. Lam, R. Sethi, and J.D. Ullman, **Compilers: Principles, Techniques, and Tools** (2nd Edition), Addison-Wesley 2007. 2. A. Appel, **Modern Compiler Implementation in C/ML/Java**, Cambridge University Press, 2004. 3. D. Grune, H.E. Bal, C.J.H. Jacobs, and K.G. Langendoen: **Modern Compiler Design**, John Wiley & Sons, Inc. 2000. 4. M.L. Scott, **Programming Language Pragmatics**, Morgan Kaufman Publishers, 2006.

Course code	CS 308N
Title of the course	Compiler Techniques
Course Category	Department core
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Automata Theory, Data Structures, and Algorithms
Objective(s)	This course will introduce the basics of a Compiler including different phases of Compiler and it's working.
Course Syllabus	• **Introduction:** Major compilation processes; Compiler phases; front end and back end partitioning. • **Lexical Analysis:** Tasks and roles of lexical analyser; Regular expressions; Deterministic finite automata; LEX – a lexical analyzer generator. • **Context-Free Grammars:** Formal grammar; Derivations; Ambiguous, unambiguous and recursive grammars; Chomsky hierarchy; Parse trees and parsing concepts. • **Syntax Analysis:** Top down parsing – recursive descent and LL(1) predictive parsers; First and Follow sets; LL(1) parse table construction; Bottom up and shift reduce parsing; LR parsing; Parse table constructions – LR(0), SLR(1) and LALR(1); YACC – a syntax analyser generator. • **Extending the Parser:** Syntax directed approach; Inherited and synthesized attributes; symbol table; Type concepts; Syntax-directed semantic analysis; Intermediate languages – three address code; Syntax directed intermediate code generation. • **Introducing Compiler Backend:** Code optimization techniques and concepts; Target code generation. • **A Complete Compiler:** The grammar specification; Scanner; parser; Code generation; Building and running the compiler. • **Practical components: D**esign and implementation of a compiler for a sufficiently rich subset of a real programming language. The compiler will be automatically generated through use of tools such as LEX, YACC and IBURG.
Suggested Books	Textbooks: 1. A.V. Aho, M.S. Lam, R. Sethi, and J.D. Ullman, ***Compilers: Principles,***

	Techniques, and Tools (2nd Edition), Addison-Wesley 2007. ISBN: 978-0321486813 2. D. Grune, H.E. Bal, C.J.H. Jacobs, and K.G. Langendoen, ***Modern Compiler Design***, John Wiley and Sons, Inc. 2000. ISBN: 978-1461446989 Reference books: 3. A. Appel, ***Modern Compiler Implementation in C/ML/Java***, Cambridge University Press, 2004. ISBN: 9780521607643 4. M. L. Scott, ***Programming Language Pragmatics***, Morgan Kaufman Publishers, 2006. ISBN: 978-0124104099

Course code	CS 310
Title of the course	Software Engineering
Course Category	Department core
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	None
Objective(s)	Understanding the creation of software applications with the view of meeting certain requirements through designing, testing and building software.
Course Outcomes	● Understanding the concept of software engineering. ● Acquiring fundamental knowledge in mathematics, computer science, programming, and computer systems. ● Understanding some ethical and professional issues that are important for software engineering.
Course Syllabus	● **Software Situation:** problems and causes; Role of Software Engineering; Software Development Paradigms. ● **Function-oriented Methodology:** System Engineering Overview; Function-Oriented Modeling Techniques; Function-Oriented Requirements Analysis; Correctness Criteria for Requirements Models; Reducing Complexity; Data Dictionary; Process Specification; Data Design; Architectural Design; Flow Analysis and Conversion Techniques; Design Refinement Measures; Procedural Design; User Interface Design. ● **Object-oriented Methodology**: Modeling of Software Requirements and Specifications with Use-Case Diagrams; Object-Oriented Modeling based on UML: Notations, Diagrams, Relationships, Modeling procedures and Applications; System Architecture; User-Interface Design; Game Interfaces and Web-based SE. Implementation: Procedural Design and Implementation. Stepwise Refinement. ● **Software Project Management:** Concerns of Management; Project Planning; Measurement and Metrics; Cost Estimation; Scheduling and Team Organization; Overview of SQA; SQA Techniques: qualitative and quantitative; Software Maintenance; Overview of Software Configuration Management; Software Configuration Items and Change Control. ● **Practical components:** ○ Students would be made to go through and experience the various phases of the Software Development Life Cycle – (1) Requirements

	Elicitation, (2) Software Design, (3) Software Development, (4) Software Testing, and (5) Software Maintenance – by working on a real project
Suggested Books	Textbooks: 1. C. Ghezzi, J. Mehdi. and M. Dino, ***Fundamentals of Software Engineering***, Prentice-Hall, 1991. ISBN: 978-0133056990 2. I. Sommerville, ***Software Engineering*** (5th Edition), Addison-Wesley, 1996. ISBN: 9780201427653 Reference books: 3. R. S. Pressman, ***Software Engineering: A Practitioner's Approach*** (6th Edition), McGraw-Hill, 2006. ISBN: 978-0073375977

Course code	CS 311
Title of the course	Parallel Computing
Course Category	Department core
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	A knowledge of programming in C, C++, or similar, the basics of data structures and the computer architecture are assumed.
Objective(s)	This course will introduce the parallel programming paradigm using different parallel programming languages.
Course outcome	Knowledge of different parallel programming languages
Course Syllabus	● Introduction to parallel computing, Amdahl's law, Computing platforms, Memory hierarchy, Shared memory, Cache coherence and synchronization. ● Interconnection networks and topologies, Routing and embeddings, Physical Organization and Communication Costs in parallel machines. ● Principles of parallel algorithm design—Concurrency, Decomposition and Mapping. ● Parallel programming using Message Passing Interface (MPI) and Shared memory platforms (i.e. OpenMP, Pthreads, CUDA etc.). ● Collective communications. Analytical modeling and metrics analysis of parallel programs. ● Practical components: ○ Experiments to support the associated theory course
Suggested Books	Textbooks: 1. A. Grama, A. Gupta, G. Karypis, and V. Kumar, ***Introduction to Parallel Computing***, Addison-Wesley, 2003, ISBN: 9780201648652 Reference books: 2. B. Chapman, G. Jost, R. van der Pas, ***Using OpenMP: Portable Shared Memory Parallel Programming***, MIT Press, 2008, ISBN: 9780262533027 3. William Gropp, Ewing Lusk, Anthony Skjellum, ***Using MPI: Portable Parallel Programming with the Message-Passing Interface***, MIT Press, 2014, ISBN: 9780262527392

Course code	CS 312
Title of the course	Foundations of Secure Computation
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Probability Theory, Discrete Mathematics, Algorithms
Objective(s)	This course will introduce the basic foundations of Secure Computation in present day distributed and computer systems.
Outcome of the Course	● The students will understand formal details and fundamental aspects of secure multiparty computation. ● The topics will enable them to understand security features of computations in distributed systems and applications of secure multiparty computation systems.
Course Syllabus	● **Introduction to Secure Computation:** Cryptography applications in Secure Communication Systems, Privacy Preserving Information Processing Computations, Abstractions and Dimensions of Secure Multiparty Computation ● **Applications of Secure Computation:** Privacy Preserving Data-mining, Secure E-auction, Yao's millionaires' problem, Privacy preserving pattern matching, Privacy preserving machine learning and its goals. ● **Secure Multiparty Computation (MPC):** Availability and confidentiality of sensitive data, Forms of function abstraction, Dimensions to study secure MPC: Protocols for Boolean circuits in asynchronous network, Protocols for arithmetic circuits in asynchronous network. ● **Overview and Basic Concepts of Abstract Algebra and Cryptography**: Characteristics of algebraic structures: Groups, Rings, and Fields, Cryptographic primitives, Real-world communication protocols: SSL, Unconditionally secure and conditionally secure schemes. ● **Secret Sharing:** Problem definition, real-world examples, additive secret sharing, Threshold secret sharing, Shamir's secret sharing, linear secret sharing, general secret sharing, Perfectly secure message transmission. ● **MPC Protocols:** Toy MPC protocol, BGW MPC protocol, Zero Knowledge Protocols, Reliable Broadcast and Byzantine agreement, Exponential Information Gathering (EIG) for Perfectly Secure Byzantine agreement.

Suggested Books	Textbooks: 1. A. Choudhury and A. Patra, ***Secure Multiparty Computation Against Passive Adversaries***, Springer, 2022. ISBN: 978-3-031-12163-0 Reference books: 2. R. Cramer, I. Djere Damgard, and J. B. Nielsen, ***Secure Multiparty Computation and Secret Sharing***, Cambridge University Press, 2015. ISBN: 9781107043053

Course code	CS 313/ MA 313
Title of the course	Computer Networks
Course Category	Department core
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Knowledge of data structures and algorithms, programming skills in C/C++/python
Objective(s)	This course will introduce computer networking protocols and performance analysis of networks.
Course outcome	Understanding the basic functionalities of computer networks
Course Syllabus	● Network Architecture and protocols. History of networking–Circuit switching and packet switching. Network performance metrics–Throughput and delay ● Application layer–HTTP, DNS, CDN, SMTP, P2P etc., ● Transport layer–UDP and TCP, Reliability and congestion control in TCP. ● Socket programming, Introduction to Network Layer. Routing protocols. Interdomain routing–BGP ● Link layer and physical layer, Performance analysis of networks. Router Architecture, Resource allocation, and QoS, Network simulation version 3 (NS3). ● Introduction to next-generation networks. ● Practical components: ○ Experimental study of application protocols such as HTTP, FTP, SMTP, using network packet sniffers and analyzers ○ Socket programming - Small exercises in socket programming in C/C++/Java. ○ Experiments with packet sniffers to study the TCP protocol. ○ Introduction to ns3 (network simulator) and small simulation exercises to study TCP behavior under different scenarios. ○ Setting up a small IP network in ns3 ○ Experiments with ns3 to study Ethernet and 802.11 wireless LAN. ○ Programming with pcap
Suggested Books	Textbooks:

	1. J. Kurose and K. Ross, ***Computer Networking, A Top-Down Approach***, Pearson Education, 8th Ed. 2022. ISBN: 978-9356061316 2. L. Peterson and B. Davie, ***Computer Networks, A Systems Approach***, Morgan Kaufmann Publishers Inc, 6th ed. 2021, ISBN: 978-0128182000 Reference books: 3. W. R. Stevens, ***Unix Network Programming: The Sockets Networking API***, Pearson Education, 3rd ed. 2017, ISBN: 978-9332549746 4. Bertsekas and Gallager, ***Data Networks***, Pearson Education 2nd ed., 2015. ISBN:978-9332550476

Course code	CS 314
Title of the course	Computer and Network Security
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematical Structures, Design and Analysis of Algorithms, Computer Networks
Objective(s)	To understand the network and computer security fundamentals.
Outcome of the course	The students will learn security mechanisms and vulnerabilities in computer systems, networks and the Internet.
Course Syllabus	● **Network Security:** Network security at application, packet Sniffing and Spoofing, Attacks on TCP Protocol, Security issues in electronic mail, IP Security, Web security, Transport layer security and Secure Socket Layer, intrusion detection, malicious software, viruses, worms and related threats, firewalls, trusted systems. ● **Computer Security:** Need for privileged programs, Set-UID mechanism: Attack Surfaces, Unsafe and safe approaches of invoking other programs, Environment variables and attack surfaces, Shellshock attack, Buffer overflow attack, Format string vulnerability: Exploitable scenarios.
Suggested Books	Textbooks: 1. Wenliang Du, ***Computer Security: A Hands-on Approach***, Amazon Digital Services, 2017, ISBN: 978-1548367947 Reference books: 2. W. Stalling, ***Cryptography and Network security Principles and Practices***, 5th Edition PHI, 2010. ISBN: 978-0136097044

Course code	CS 315
Title of the course	Introduction to Complexity Theory
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Algorithms, Formal Languages and Automata Theory, Discrete Mathematics
Objective(s)	This Course is designed to provide an overview of Turing machine, the details of the complexity classes and their relationships.
Course Outcomes	Students will be able to understand computational complexity and their significance.
Course Syllabus	• **Introduction to the Turing Machine:** Definition of Turing Machines, Examples, Deterministic and non-deterministic Turing machines, Other variants of Turing machine, The Definition of Algorithm • **Decidability and undecidability:** Reducibility, Undecidable Problems from Language Theory, Mapping Reducibility • **P, NP and NP-completeness:** The Class P, Class NP, NP-hard, NP-completeness, Reducibility between problems, Discussions on different NP-complete problems such as satisfiability, clique, vertex cover, independent set, set cover, TSP, etc. • **Space complexity:** Savitch's Theorem, The Class PSPACE, PSPACE-completeness, Class L and NL, NL-completeness, NL equals coNL
Suggested Books	Textbooks: 1. S. Arora and B. Barak, ***Computational Complexity: A Modern Approach***, Cambridge University Press, 2009. ISBN: 978-0521424264 2. M. Sipser, ***Introduction to the Theory of Computation***, 3rd eds., Cengage Learning, 2012. ISBN: 978-8131525296 Reference Books: 3. S. Rudich and A. Wigderson, ***Computational Complexity Theory***, 1st eds., American Mathematical Society, 2004. ISBN: 978-0821828724

Course code	**CS 317**
Title of the course	**Introduction to Internet of Things**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Basic arduino programming and networking
Objective(s)	This course will introduce the basic components of IoT and their interdependencies, deployment models, and fundamental concepts of IoT networking.
Course outcome	● Understanding basic concepts and functionalities of IoT devices and networking ● Developing IoT-based prototype
Course Syllabus	● Introduction to IoT; Sensing, Actuation, Basics of Networking; Basics of Networking; ● Communication Protocols; Sensor Networks; Machine-to-Machine Communications, Interoperability in IoT ● Arduino Programming; Integration of Sensors and Actuators with Arduino; Introduction to Python Programming, ● Implementation of IoT with Raspberry Pi, Implementation of IoT with Raspberry Pi ● Cloud Architecture and its functionalities; Introduction to edge architectures and its functionalities ● Smart Cities and Smart Homes; Connected Vehicles, Smart Grid, Industrial IoT; Case Study: Agriculture, Healthcare, Activity Monitoring
Suggested Books	Textbook: 1. S. Misra, A. Mukherjee, A. Roy, ***Introduction to IoT***, Cambridge University Press, 2022. ISBN: 978-1108959742 Reference book: 2. A. Bahga and V. Madisetti, ***Internet of Things: A Hands-On Approach***, Orient Blackswan Private Limited, 2015. ISBN: 978-8173719547

Course code	**CS 319**
Title of the course	**Foundations of Cryptography**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematical Structures, Design and Analysis of Algorithms, Computer Networks
Objective(s)	● To understand the basic foundations of cryptography, understand the encryption and authentication protocols with security proofs. ● The students will study block ciphers, stream ciphers, hash functions and public key cryptography.
Outcome of the course	● The students will learn about different proving models, indistinguishability tests about security of encryption algorithms, authentication algorithms, and hashing algorithms. ● Along with gaining knowledge about security bounds, the students will learn some number theory and algebra, wherever required.
Course Syllabus	● **Introduction:** Classical ciphers, Cryptanalysis techniques: linear and differential cryptanalysis. ● **Number Theory:** Euclidean Algorithm, Chinese Remainder Theorem, Primality Testing algorithms, Factoring algorithms, Algebraic Structures: Groups, Rings and Fields. ● **Shannon's theory:** Concept of perfect secrecy, Entropy, Key equivocation, Unicity Distance, Perfect cipher, Ideal Cipher. ● **Symmetric-key Cryptography:** Pseudorandomness, Stream ciphers, Block ciphers, Data Encryption Standards, Advanced Encryption Standards, Modes of operation ● **Hash-functions:** Data Integrity, Merkle-Damgard construction, Message Authentication Codes. ● **Public-key Cryptography:** RSA, Discrete log problem, DiffieHellman key exchange protocol, Signatures schemes, Public key Infrastructure, Digital certificates.
Suggested Books	Textbooks: 1. D. R. Stinson, ***Cryptography: Theory and practices***, 3rd Edition, CRC Press, 2006. ISBN: 978-1584885085 2. J. Katz and Y. Lindell, ***Introduction to Modern Cryptography***, Chapman and Hall/CRC, 2020. ISBN: 978-0815354369 Reference books: 3. A. J. Menezes, P. Oorschot, and S. Vanstone, ***Handbook of Applied Cryptography***, CRC Press, 1997. ISBN: 9781138385979

	4. W. Stalling, ***Cryptography and Network security Principles and Practices***, 5th Edition Pearson Education, 2017. ISBN: 978-9332585225

Course code	**CS 321**
Title of the course	**Introduction to Big Data Analysis**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Linear Algebra, Calculus, Data Structures, Algorithms
Objective(s)	To help students to learn and understand the terminologies and core concept behind big data problems, applications, and modern big data computing technologies.
Course Outcomes	Students would be familiar with real-world challenges associated with Big Data.
Course Syllabus	● **Introduction:** Types of Data under Big Data, Characteristics of Big Data, 5 V's of Big Data (Velocity, Volume, Value, Variety and Veracity), Challenges and Applications of Big Data. ● **Technologies:** Apache Spark, HDFS, YARN, Introduction to MapReduce, MapReduce Programming Model with Spark, MapReduce Example: Word Count, Page Rank etc. ● **Storage Platforms:** Introduction to HBase, HBase Internals, Hive, Pig, Ozie, Introduction to Big Data Streaming Platforms for Fast Data, Introduction to Spark Streaming, Kafka. ● **Big Data Machine learning:** Machine Learning with Spark, Introduction to Spark MLlib, Cluster Analysis, Association Analysis.
Suggested Books	Textbooks: 1. R. Kamal and P. Saxena, ***Big Data Analytics, Introduction to Hadoop, Spark, and Machine-Learning***, McGraw Hill Education, 2019. ISBN: 978-9-3531-6496-6. Reference Books: 2. C. Lam, ***Hadoop in Action***, Manning Publications, 2010. ISBN: 978-1-9351-8219-1. 3. H. Karau, A. Konwinski, P. Wendell, and M. Zaharia, ***Learning Spark: Lightning-Fast Big Data Analysis***, O'Reilly, 2015. ISBN: 978-1-4493-5862-4.

Course Code	**CS 352**
Title of the Course	**Computer Graphics and Visualization Lab**
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Computer Programming, Data Structures and Algorithms
Scope of the course	This Course is designed to provide an introduction to the theory and practice of computer graphics and an insight of modern computer graphics systems. Students will understand the basic principles of computer graphics primitives and able to design application specific computer graphics program. This course will also provide familiarity with key algorithms for modelling and rendering graphical data.
Course Syllabus	Assignments based on of applications of computer graphics and visualizations in the fields such as 3D-modeling of architectural and mechanical design; Creating 3D games; Creating 3D models from segmented volume data; Financial data visualization.
Suggested Books	1. D. Hearn, M. P. Baker, ***Computer Graphics. C Version***, Pearson Education , 2nd Eds, 1997 2. D. Hearn, M. P. Baker, ***Computer Graphics with OpenGL***, Pearson Education India , 3rd Eds, 2004 3. F.S. Hill. ***Computer Graphics Using Open GL***. Prentice Hall. 2001 4. John F. Hughes, Andries van Dam, James D. Foley, Morgan McGuire, Steven K. Feiner, David F. Sklar, Kurt Akeley, ***Computer Graphics, Principles and Practice***, Addison Wesley, 3rd Eds, 2014. 5. W. Schroeder, K. Martin, and B. Lorensen, ***The Visualization Toolkit***, (2nd Edition), Prentice-Hall, Inc., 1998. 6. M. K. Pakhira, ***Computer Graphics, Multimedia and Animation***, PHI, 2nd Eds, 2010

Course Code	**CS 353**
Title of the Course	**Operating Systems Lab**
Credit Structure	L-T-P-Credits 0-0-3- 1.5
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	Knowledge of Computer Programming
Scope of the course	
Course Syllabus	OS Programming prerequisites: Familiarities with IPC facilities, IPC identifiers, IPC keys, Message queues and their internal and user data structures, System calls related to IPC, Semaphore and Shared memory. (06 hours ≈2 labs). CPU scheduling: Simulation programs for long-term, short-term and medium term schedulers, Simulation for the maintenance of various scheduling queues such as ready, I/O, blocked etc., Implementations of different scheduling algorithms such as FCFS, SJF, Priority scheduling (pre-emptive and Non pre-emptive), Round robin, multilevel feedback queue scheduling and their performance evaluations. (12 hours ≈4 labs). Concurrent Processing and Concurrency Control: Simulation of updating four processes PCBs with shared memory, Implementation of interprocess communication using simulated semaphore through i) shared memory, ii) synchronized producer-consumer problem iii) pipes and message passing (asynchronous and synchronous). Concurrence control with pipes socket for iterative and concurrent servers (12 hours ≈4 labs). File Systems Implementation: creating, removing, accessing and protection and error handling of EXT2 FS, Registering the virtual file system in Kernel, accessing superblock information. (06 hours ≈2 labs).
Suggested Books	**1.** Linus Programmer's Guide documentation **2.** UNIX System V and Related Utilities under Linux

Course code	**CS 353N**
Title of the course	**Operating Systems Lab**
Course Category	Department core
Credit Structure	L - T - P - Credits 0-0-2-1
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	NA
Objective(s)	This course will introduce the basic components of operating systems and functionalities.
Course outcome	Understanding basic functionalities of operating system for efficient performance of the processes
Course Syllabus	• **OS Programming prerequisites:** Familiarities with IPC facilities, IPC identifiers, IPC keys, Message queues and their internal and user data structures, System calls related to IPC, Semaphore and Shared memory. • **CPU scheduling:** Simulation programs for long-term, short-term and medium term schedulers, Simulation for the maintenance of various scheduling queues such as ready, I/O, blocked etc., Implementations of different scheduling algorithms such as FCFS, SJF, Priority scheduling (preemptive and non-preemptive), Round robin, multilevel feedback queue scheduling and their performance evaluations. • **Concurrent Processing and Concurrency Control:** Simulation of updating processe PCBs with shared memory, Implementation of interprocess communication using simulated semaphore through (i) shared memory, (ii) synchronized producer-consumer problem, (ii) Pipes and message passing (asynchronous and synchronous). Concurrence control with pipes socket for iterative and concurrent servers • **File Systems Implementation:** creating, removing, accessing, protecting and error handling of EXT2 FS, Registering the virtual file system in Kernel, accessing superblock information.
Suggested Books	Textbooks: 1. A. Silberschatz, P.B. Galvin, and G. Gagne, ***Operating System Principles***, 7th edition, John Wiley, 2005. ISBN: 9788126509621 2. A. Silberschatz, P.B. Galvin, and G. Gagne, ***Operating System Concepts***, 9th edition, Wiley, 2018. ISBN: 978-1-118-06333-0

	Reference books: 3. W. Stallings, ***Operating Systems: Internals and Design Principles***, 5th edition, Pearson Education, 2005. ISBN: 978-0-13-467095-9

Course Code	CS 354
Title of the Course	Computational Intelligence Lab
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Computer Programming, Data structure, Discrete Structure, Design and Analysis of Algorithm
Scope of the course	
Course Syllabus	AI programming : Prolog, LISP, Experiments to support the associated theory course that demonstrate the different applications of Neural, fuzzy, evolutionary and hybrid model; Minor project based on real life applications such as Functional approximation; Time-series prediction; Pattern recognition; Data compression; Control applications, Optimization etc.
Suggested Books	Same as the associated theory course CS 304N: Computational Intelligence

Course code	CS 354N
Title of the course	Computational Intelligence Lab
Course Category	Department core
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Computer Programming, Data structure, and Design and Analysis of Algorithm
Objective(s)	Basics of machine learning techniques
Course Outcomes	Understanding of machine learning techniques and implementation
Course Syllabus	● **AI programming:** Prolog, LISP, Experiments to support the associated theory course that demonstrate the different applications of Neural, fuzzy, evolutionary and hybrid model; ● **Implementation:** Minor project based on real life applications such as Functional approximation; Time-series prediction; Pattern recognition; Data compression; Control applications, Optimization etc.
Suggested Books	Textbooks: 1. S. Russell and P. Norvig, ***Artificial Intelligence: A Modern Approach***, Prentice Hall Series in AI, 1995. ISBN: 978-9332543515 2. E. Rich and K. Knight, ***Artificial Intelligence***, Tata McGraw Hill, 1992. ISBN: 978-0-07-067816-3 Reference books: 3. J.S.R.J ang, C.T. Sun and E. Mizutani, ***Neuro-Fuzzy and Soft Computing***, Prentice Hall and Pearson Education, 2004. ISBN: 978-9332549883 4. D.E. Goldberg, ***Genetic Algorithms: Search, Optimization and Machine Learning***, Addison Wesley, 1989. ISBN: 9781584883883 5. S. Rajasekaran and G.A.V. Pai, ***Neural Networks, Fuzzy Logic and Genetic Algorithms***, Prentice Hall, 2003. ISBN: 9788120321861 6. R. Eberhart, P. Simpson and R. Dobbins, ***Computational Intelligence - PC Tools***, AP Professional, 1996. ISBN: 978-0122286308

Course Code	**CS 355**
Title of the Course	**Computer Architecture Lab**
Credit Structure	L-T-P-Credits 0-0-3- 1.5
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	A course in Logic Design
Scope of the course	
Course Syllabus	**MIPS Programming through SIMPS:** Familiarities with architecture of RISC Computer R2000/R3000 proposed in MIPS Systems. MIPS Assembly language programming for instruction formats, addressing mechanism, microprogramming to transfer data between register-register, memory-register and architectural programming. (12 hours ≈4 labs) **Architecture-Level Design with Verilog:** Familiarize architecture-level design and synthesis of different components in arithmetic and logic unit. Verilog programming to design basic computing units such as adder, multiplier, BCD converter, Comparator etc. Experiment for datapath synthesis, connecting memory, buffer, external ports and different components in an application specific processing unit. (12 hours ≈4 labs) **Synthesis of a CPU Architecture:** Familiarize the design aspects of a CPU to realize the design in a FPGA kit. Designing a CPU with a selected specification at architectural-level using Verilog, and finally, realizing the architecture in a FPGA kit followed by testing the correctness of the realization. (12 hours ≈4 labs)
Suggested Books	Same as CS 305

Course Code	**CS 356**
Title of the Course	**Computer Networks Lab**
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Data Structures and Algorithms
Scope of the course	
Course Syllabus	(a) Experimental study of application protocols such as HTTP, FTP, SMTP, using network packet sniffers and analyzers such as Wireshark. (b) Socket programming - Small exercises in socket programming in C/C++/Java. (c) Experiments with packet sniffers to study the TCP protocol. 3-way handshake for connection setup, timer behavior, congestion control behavior. (d) Introduction to ns3 (network simulator) and small simulation exercises to study TCP behavior under different scenarios. (e) Setting up a small IP network in ns3 - configure interfaces, IP addresses and routing protocols to set up a small IP network. Study dynamic behavior using packet sniffers. (f) Experiments with ns3 to study behavior (especially performance of link layer protocols such as Ethernet and 802.11 wireless LAN. (g) Programming with pcap - small example with packet generator using pacp library
Suggested Books	Same as CS 306: Computer Networks

Course Code	**CS 357**
Title of Course	**Optimization Algorithms and Techniques Lab**
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Department	Computer Science and Engineering
Pre-requisite, if any	Data Structures and Algorithms
Scope of the course	
Course Syllabus	Experiments and assignments based upon techniques discussed in CS 307. These are summarized below. * Understanding of Matlab/ Scilab via implementation of Newton's method for solving non-linear system of equations as well as numerical integration. * Analyzing convexity of functions numerically. * Implementation and analysis of Multi-dimensional Unconstrained Optimization algorithms (Steepest Descent, Newton, Gauss-Newton, Quasi-Newton, Conjugate Gradients etc.). * Implementation and analysis of One-dimensional Unconstrained Optimization algorithms (Dichotomous, Quadratic Interpolation, Cubic Interpolation etc.). * Implementation and analysis of Simplex and Interior Point Methods for Linear Program. * Implementation and analysis of Sequential Quadratic Program for solving general Constrained Optimization problem.
Suggested books	Same as the associated theory course CS 307

Course code	**CS 357N**
Title of the Course	**Optimization Algorithms and Techniques Lab**
Course Category	Department core
Credit Structure	L - T - P - Credits 0-0-2-1
Name of the Concerned Discipline	Computer Science and Engineering
Pre-requisite, if any	Knowledge of Data Structures and Algorithms
Objective(s)	This is an introductory course in the field of mathematical optimization.
Course Outcomes	At the end of the course, students will know ● The Basics of Optimization, ● Unconstrained and Constrained Optimization, and ● Linear and Quadratic Programming.
Course Syllabus	● Understanding of Matlab/ Scilab via implementation of Newton's method for solving non-linear system of equations as well as numerical integration. ● Analyzing convexity of functions numerically. ● Implementation and analysis of Multi-dimensional Unconstrained Optimization algorithms (Steepest Descent, Newton, Gauss-Newton, Quasi-Newton, Conjugate Gradients etc.). ● Implementation and analysis of One-dimensional Unconstrained Optimization algorithms (Dichotomous, Quadratic Interpolation, Cubic Interpolation etc.). ● Implementation and analysis of Simplex and Interior Point Methods for Linear Program. ● Implementation and analysis of Sequential Quadratic Program for solving general Constrained Optimization problem.
Suggested Books	Textbooks: 1. J. Nocedal and S. J. Wright, ***Numerical Optimization***, 1st Edition, Springer, 2006. ISBN: 78-1-4939-3711-0 Reference books: 2. A. Antoniou and W.-S.g Lu, ***Practical Optimization: Algorithms and Engineering Applications***, 2nd Edition, Springer, 2021. ISBN: 978-1-0716-0843-2

Course Code	**CS 358**
Title of the Course	**Compiler Techniques Lab**
Credit Structure	L-T-P-Credits 0 -0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Same as the associated theory course
Scope of the course	
Course Syllabus	Design and implementation of a compiler for a sufficiently rich subset of a real programming language. The compiler will be automatically generated through use of tools such as LEX, YACC and IBURG.
Suggested Books	1. J.R. Levine, T. Mason, and D. Brown, **LEX and YACC**, O'Reilly & Associates, 1990

Course Code	**CS 401** [From AY 2010-11 to 2013-14]
Title of the Course	**Soft Computing**
Credit Structure	L-T-P-Credits 3-0-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	**Introduction:** Artificial neural networks: feed-forward, recurrent and multi-layer architectures; Supervised and unsupervised learning; Characteristics: adaptability, fault tolerance, generalization; limitations of neuro-computing. **Perceptron:** Linear classifiers; Simple perceptron; Perceptron learning algorithms; ADALINE; MADALINE; Limitation of perceptron dichotomizer. **Multi-Layer Perceptron:** Gradient decent scheme for error minimization; Generalized delta learning rule; Back-propagation learning for multi-layer networks; Multi-layer perceptrons for multi-dimensional functional mappings. **Associated Memory Networks:** Auto-association; Hetero-association; Linear associative networks: Hebbian learning, perfect recall, cross-talk; Bidirectional associative memory; Brain-State-in-a-Box network. **Hopfield Networks:** Binary Hopfield network: basic structure, asynchronous updating, convergence, associative memory; Continuous-valued Hopfield network. Advantages and limitations. **Kohonen Networks:** Self-organizing networks; Similarity measures; Kohonen's winner-take-all network; Geometrical interpretation of Kohonen's learning; Functional specificity of human brain, Kohonen's self-organizing feature map algorithm; Conscience algorithm. **Adaptive Resonance Theory (ART):** ART and stability-plasticity dilemma; ART-1 architecture and algorithm: search, comparison and recognition phases, effect of vigilance. **Radial Basis Function Networks:** Radial Basis Function Networks: radial basis vs. linear basis, Gaussian basis functions, K-means learning, LMS algorithm, comparison with Multi-Layer Perceptron networks. **Support Vector Machines (SVM):** Optimal hyperplane for linear separability, quadratic optimization, SVM for pattern recognition, different kernels for hidden-layer, optimal design of SVM. **Fuzzy Neural Networks and Genetic Algorithms:** Fuzzy sets - Basic Definition; Fuzzy-set-theoretic Operations – Member Function Formulation and Parameterization – Fuzzy Rules and Fuzzy Reasoning, Fuzzy If-Then Rules Fuzzy-neural networks; Neuro-

	fuzzy systems; Genetic algorithms: selection schemes, operations, hybrid algorithms.
Suggested Books	1. J.S.R.J ang, C.T. Sun and E. Mizutani, "**Neuro-Fuzzy and Soft Computing"**, Prentice Hall of India and Pearson Education, 2004. 2. D.E. Goldberg, "**Genetic Algorithms: Search, Optimization and Machine Learning"**, Addison Wesley, New York, 1989. 3. S. Rajasekaran and G.A.V. Pai**, "Neural Networks, Fuzzy Logic and Genetic Algorithms"**, Prentice Hall of India, 2003. 4. R. Eberhart, P. Simpson and R. Dobbins, **"Computational Intelligence - PC Tools"**, AP Professional, Boston, 1996.

Course Code	**CS 401 / CS 601** [from AY 2014-15 to AY 2023-24]
Title of the Course	**Soft Computing**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department/Discipline	Computer Science and Engineering
Pre–requisite, if any	Discrete Mathematical Structures, Design and Analysis of Algorithms, Computational Intelligence
Scope of the Course	After having basic knowledge of artificial intelligence related to neural, fuzzy and evolutionary approaches, advancements in different areas are to be covered with working in a specific domain. This is by taking a case study to come up with the implementation and results.
Course Syllabus	**Review on Mathematical and theoretical methods on soft computing**: neural networks. RBF structures. Self- organizing networks and methods. Fuzzy logic. Support vector machines and kernel methods. Evolutionary algorithms. **Hybrid Intelligent Systems:** Neuro-fuzzy systems. Neuro-Genetic systems, Evolving neural systems. Neuro-swarm. Hybridization with novel computing paradigms: Quantum computing, DNA computing, membrane computing. Neural dynamic logic and other methods, etc. **Learning and adaptation for novel:** Adaptive systems. Imitation learning. Reconfigurable systems. Supervised, unsupervised, Semi-supervised, reinforcement and statistical algorithms. Stability and convergence analysis. **Applications:** Image and signal processing. Ambient intelligence. process control, and manufacturing. Biometry and bioinformatics. Data mining. Internet modeling, communication and networking. Intelligent systems in education. Human–robot interaction. Time series analysis and prediction etc.
Suggested Books	**Book:** 1. Jang, Roger and Mizutani, "Neuro-Fuzzy and Softcomputing: A Computational Approach to learning and Machine Intelligence", Pearson. 2. R. John and Ralph Birkenhead, SoftComputing Techniques and Applications (Advances in Intelligent and Softcomputing), 2000, Springer-Verlag. 3. F.O. Karray, C. W. De Silva, SoftComputing and Intelligent System Design: Theory, Tools and Applications, Addison Wesley; 1st Ed. 2004. **Other References:** 1. IEEE Transactions on Fuzzy Systems 2. ACM Transactions on Knowledge Discovery from Data (TKDD) 3. The journal of pattern recognition society, ELSEVIER

	4. The journal of Neurocomputing, ELSEVIER 5. IEEE Transactions on Evolutionary Computation 6. IEEE Transactions on Neural Networks Learning Algorithms 7. Other web resources will be posted on the course website from time to time.

Course Code	**CS 402** [**CS 309** from AY 2015-16 onwards]
Title of the Course	**Parallel Computing**
Credit Structure	L-T-P-Credits 3-0-0-3 / 2-0-2-3 [for AY 2014-15]
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	**Introduction to Parallel Algorithms:** Basic schemes for parallelization: list ranking. NC class. Parallel Context Free Grammar Parsing Algorithms. **Distributed Computing:** absence of global states; causal ordering of events. **Distributed architectures:** shared memory and message passing, Programming Models such as PVM; MPI; Linda; ORCA, Distributed algorithms: mutual exclusion, consensus, leader election. Clock synchronization, distributed termination. **Fault Tolerance:** fail-stop and byzantine models.
Suggested Books	1. A. Gibbons, and W. Rytter, **Efficient Parallel Algorithms**, Cambridge University Press, 1989, ISBN: 0521388414. 2. H. Attiya and J. Welch, **Distributed Computing: Fundamentals, Simulations, and Advanced Topics**, McGraw-Hill Inc. New York, 1998. 3. 3. 3. G. F. Colouris, and J. Dollimore, **Distributed Systems: Concepts and Design**, Addison Wesley, 1988. 4. N. Lynch, **Distributed Algorithms**, Morgan Kaufmann, 1996. 5. S. Mullender (Ed.), **Distributed Systems** (2nd Edition), Addison Wesley, 1993. 6. T. Gerard, **Introduction to Distributed Algorithms**, Cambridge University Press, Cambridge, 1994. 7. M. Raynal, **Distributed Algorithms and Protocols**, Wiley, Chichester, 1988. 8. V.C. Barbosa, **An Introduction to Distributed Algorithms**, MIT Press, 1996.

Course code	**CS 403/ CS 603**
Title of the course	**Machine Learning**
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre-requisite, if any	Artificial Intelligence/Computational Intelligence
Scope of the course	This course provides a broad introduction to machine learning, datamining, and statistical pattern recognition. Topics include: (i) Supervised learning (parametric/non-parametric algorithms, support vector machines, kernels, neural networks). (ii) Unsupervised learning (clustering, dimensionality reduction, recommender systems, deep learning). (iii) Best practices in machine learning (bias/variance theory_ innovation process in machine learning and AI). The course will also draw from numerous case studies and applications, so that candidate's also learn how to apply learning algorithms to build different intelligent systems.
Course Syllabus	Introduction, Machine Learning and AI, Motivations for Studying ML, Supervised and Unsupervised learning, Linear prediction, Maximum likelihood Regularizers, basis functions and cross-validation, Optimisation, Linear and Logistic Regression, Gaussian Discriminant Analysis, Support Vector Machines, Decision Trees, Neural networks architectures and its advances, Ensemble Methods, Clustering, Naive Bayes, Bayesian Statistics, K-Means, Gaussian Mixture Models, Learning Theory, Model Selection.
Suggested Books	1. C. M. Bishop, ***Pattern Recognition and Machine Learning,*** Springer, Heidelberg, 2006, 978-0-387-31073-2 2. T. Mitchell, ***Machine Learning***, McGraw Hill, 1997 (new chapters on line, 2006), New York, 1997, 978—0071154673 3. Duda, Hart and Stork, ***Pattern Classification*** *(2nd ed.),* Wiley Interscience, US, 2000, 978-8126511167

Course Code	**CS 404/ EE 304**
Title of the Course	**Digital Signal Processing**
Credit Structure	L-T-P-Credits 3-1-0-4
Name of the Concerned Department	Electrical Engineering
Pre–requisite, if any	Signals and Systems Course
Scope of the course	
Course Syllabus	Discrete time signals: Sequences; representation of signals on orthogonal basis; Sampling and reconstruction of signals; Discrete systems: attributes, Z-Transform, Analysis of LSI systems, Frequency analysis, Inverse Systems, Discrete Fourier Transform (DFT), Fast Fourier Transform algorithm, Implementation of Discrete Time Systems Design of FIR Digital filters: Window method, Park-McClellan's method. Design of IIR Digital Filters: Butterworth, Chebyshev and Elliptic Approximations; Lowpass, Bandpass, Bandstop and High pass filters. Effect of finite register length in FIR filter design. Parametric and non-parametric spectral estimation. Introduction to multirate signal processing. Application of DSP to Speech and Radar signal processing.
Suggested Books	1. A.V. Oppenheim and Schafer, **Discrete Time Signal Processing**, Prentice Hall, 1989. 2. J.G. Proakis and D.G. Manolakis, **Digital Signal Processing: Principles, Algorithms And Applications**, Prentice Hall, 1997. 3. L.R. Rabiner and B. Gold, **Theory and Application of Digital Signal Processing**, Prentice Hall, 1992. 4. J.R. Johnson, **Introduction to Digital Signal Processing**, Prentice Hall, 1992. 5. D.J. DeFatta, J.G. Lucas, and W.S. Hodgkiss, **Digital Signal Processing**, John Wiley & Sons, Singapore, 1988.

Course Code	**CS 406 / CS 606**
Title of the Course	**Data Mining and Data Warehousing**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre-Requisite, if any	Data Base & Information Systems
Scope of the course	
Course Syllabus	**Data Warehouse and OLAP Technology:** Data warehousing Definition, usage and trends, Data marts, Metadata, Multidimensional data model, Data cubes, Schemas for Multidimensional Database: stars, snowflakes and fact constellations, Data warehouse architecture, OLTP and OLAP, types of OLAP servers: ROLAP, MOLAP, 3- Tier data warehouse architecture, Data warehouse implementation, computation of data cubes, indexing OLAP data, processing OLAP queries. **Data Mining:** Data mining definition & task, data preprocessing, data mining functionalities: Characterization and Discrimination, Mining frequent patterns, Frequent itemset mining methods, associations, and Correlations, Classification and Predictions, Cluster Analysis, Outlier Analysis, Evolution Analysis **Mining complex data objects:** Spatial databases, Multimedia databases, Time series and Sequence data, mining Text Databases and mining Word Wide Web, Applications and Trends in Data Mining
Suggested Books	**1.** Jiawei Han and Micheline Kamber, **Data Mining: Concepts and Techniques**, Second Edition, Elsevier Publication. 2. M. H.Dunham, **Data Mining: Introductory and Advanced Topics**, Pearson Education 2004.

Course Code	**CS 407**
Title of the Course	**Peripherals and Interfaces**
Credit Structure	L-T- P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre–requisite, if any	Computer Architecture
Scope of the course	This course deals with the various aspects of hardware software interfacing with peripherals and associated devices. The course covers the fundamentals of various peripheral devices, its programming through assembly language and architecture. Further, it provides the an avenue for learning concepts of microprocessors, microcontrollers, interrupts and memory access mechanisms.
Course Syllabus	**Basics of Microprocessor:** Design, Memory Subsystems, System Resources, Types and Interrupt handling, 8085 Architecture and its programming, 8086 Architecture and its programming, DMA channel, I/O port addresses. I/O buses, Local bus, DMA controller, PCI, ADC/DAC interfacing with microcontrollers/microprocessors. GPUs, USB, Bluetooth, 8255 interfacing, RAID. Video Hardware, Video display technologies, Introduction to serial communication, 8253/8254 programmable timer and interval counter. I/O Interfaces, USB Basic and Driver model Testing of serial and parallel port, USB mouse/keyboard interfaces. Interrupt Controller, Video/Graphics of Modern Desktop Board, Concepts of Network Interface Card, Design and Integration of Peripheral devices to a computer system as a Case Study.
Suggested books	1. Douglas V. Hall. **Microprocessor and Interfacing: Programming and Hardware.** McGraw Hill Inc. 2. Ramesh S. Gaonkar, **Microprocessor Architecture, Programming and Application with the 8085.** 5th edition, Penram India 3. Stuart R. Ball. **Analog Interfacing to Embedded Microprocessors.** Elsevier, 2014.

Course Code	**CS 408**
Title of the Course	**Algorithms for Convex Programming**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre-Requisite, if any	
Scope of the course	
Course Syllabus	
Suggested Books	

Course Code	**CS 409 / CS 609**
Title of the Course	**Advanced Topics in Database Management Systems**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	Data Structures and Algorithms and Database and Information Systems
Scope of the course	
Course Syllabus	**Advanced Data Models:** Enhanced Relational System, Object-Oriented Data Model, Spatial and Temporal Databases, Multimedia Databases. **Query Processing and Optimization:** Query Interpretation and Equivalence Expressions, Cost Estimate in Query Optimization, Semantic Query Optimization. **Transaction Processing and Concurrency Control:** Properties of Transactions, Schedules and Serializability of Schedules, Transaction Failures and Recoverability, High Performance Transaction Systems. **Distributed Databases:** Design of Distributed Databases, Distributed Query Processing, Deadlock Handling, Concurrency Control and Recovery. **Database Security and Authorization:** Database Security Issues, Security and Integrity Violations, Multilevel Security, Discretionary and Mandatory Access Control, Statistical Database Security.
Suggested Books	1. R. Elmasri and S. Navathe, **Fundamentals of Database Systems** (3rd Ed), Benjamin Cummings, 2002. 2. H. F. Korth and A. Silberschatz, **Database System Concepts** (3rd Ed.), McGraw Hill Inc., 2003 3. C. Zaniolo, S. Ceri, C. Faloutsos, Richard T. Snodgrass, V.S. Subrahmanian, R. Zicari, **Advanced Database Systems**, Morgan Kauffmann, 2002

Course Code	**CS 410**
Title of the Course	**Genetic Algorithms**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	Optimization Algorithms and Techniques
Scope of the course	
Course Syllabus	**Evolutionary Computations:** Biological background, Canonical GA framework, Basic Terminologies, Formulation of Optimization problems into GA framework. **Variations of GAs:** Binary Coded GAs and its variations such as Micro GA, Messy GA, Greedy GA etc., Real Coded GAs, Permutation Encoding GA etc. **GA operators:** Selection, Reproduction, Crossover, Mutation etc. Convergence criteria, Mathematical Construction of Genetic Operators, Schema Theorem of John Holland. Advanced Operators and Techniques in GA: Diploidy and Multiploidy, Inversion and Reordering, Niche and Speciation, Segregation and Translocation. **Multi-Objective GAs:** Non Pareto and Pareto-based GAs, MOGA, NSGA, Niched Pareto Genetic Algorithm. Practice of GA with some real-life problems and GA Programming: Traveling Salesman Problem, Word Matching problem, Topological Planning in Wireless Network, Placement and Routing problem in VLSI Design, Image Processing and Pattern Recognition.
Suggested Books	1. D. E. Goldberg, **Genetic Algorithms in Search , Optimization & Machine Learning**, Pearson Education, 2000. 2. K. Deb, **Multi-Objective Optimization using Evolutionary Algorithms,** John-Wiley & Sons, Ltd. Chichester, 2001. 3. T. Back, David B. Fogel, Z. Michalewicz, **Handbook of Evolutionary Computation**, Oxford University Press, 1999. 4. M. Mitchell, **An Introduction to Genetic Algorithms** (3rd Ed) Bradford Book, 1998.

Course Code	**CS 411/ CS 611**
Title of the Course	**Advanced Algorithms**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	Data Structures and Algorithms and Design and Analysis of Algorithms
Scope of the course	
Course Syllabus	**Advanced Solutions to Basic Data Structuring Problems:** Binomial heaps and Fibonacci heaps, Red-Black tree, Splay tree, van Emde Boas Priority Queues, Dynamic Data Structures for Graph Connectivity/Reachability. **Bit Tricks Techniques:** Word-level Parallelism, Trans dichotomous Model, O(n) and O (log n) Integer Sorting. **String Algorithms:** Rabin-Karp Fingerprinting Algorithm, Suffix Trees. **Maximum Flows:** Augmenting Paths and Push-Relabel Methods, Minimum Cost Flows, Bipartite Matching. **Linear Programming:** Formulation of Problems as Linear Programs, Duality, Simplex, Interior Point, and Ellipsoid Algorithms. **Online Algorithms:** Ski Rental, River Search Problem, Paging, The k-Server Problem, List Ordering and Move-to-Front. **Approximation Algorithms:** One Way of Coping with NP-Hardness, Greedy Approximation Algorithms, Dynamic Programming and Weakly Polynomial-Time Algorithms, Linear Programming Relaxations, Randomized Rounding, Vertex Cover, Wiring and TSP. **Fixed-Parameter Algorithms:** Parameterized Complexity, Kernelization, Vertex Cover, Connections to Approximation. **Parallel Algorithms:** PRAM. Pointer Jumping and Parallel Prefix. Tree Contraction. Divide and Conquer. Randomized Symmetry Breaking. Maximal Independent Set. **External-Memory Algorithms:** Accounting for the Cost of Accessing Data from Slow Memory. Sorting. B-trees. Buffer Trees. Cache-oblivious Algorithms for Matrix Multiplication and Binary Search. **Computational Geometry:** Convex Hull. Line-segment Intersection. Sweep Lines. Voronoi Diagrams. Range Trees. Seidel's Low-dimensional LP Algorithm.
Suggested Books	1. T. Cormen, C. Leiserson, R. Rivest, and C. Stein. **Introduction to**

	Algorithms. (3rd Ed). MIT Press, McGraw-Hill, 2010. 2. R. Motwani and P. Raghavan, **Randomized Algorithms**, Cambridge University Press, 1995. 3. V. V. Vazirani, **Approximation Algorithms**, Springer. 2001. 4. Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin, **Network Flows: Theory, Algorithms, and Applications**, Prentice Hall, 1993.

Course Code	**CS 412/ CS 612**
Title of the Course	**Pattern Recognition**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre-Requisite, if any	Basics of probability theory, Programming
Scope of the course	This course aim to cover the basic concepts for analyzing patterns and their preprocessing techniques. It also aims to give exposure to various learning algorithms and their applications to various real life applications.
Course Syllabus	1. **Basics of pattern recognition:** Definitions, data sets for pattern recognition, representations of patterns and classes, metric and non-metric proximity measures, feature extraction, statistical and syntactic pattern recognition 2. **Bayesian decision theory:** Classifiers, discriminant functions, decision surfaces, normal density and discriminant functions, discrete features 3. **Parameter estimation methods:** Maximum-likelihood estimation, expectation-maximization method, Bayesian estimation, Gaussian mixture models 4. **Non-parametric techniques:** Density estimation using Parzen-window method, K-nearest neighbor method, nearest neighbor classifier 5. **Dimension reduction methods:** Lineardiscriminant analysis (LDA), principal component analysis (PCA) 6. **Linear discriminant function based classifiers:** Perceptron, support vector machines (SVM) 7. **Non-metric methods for pattern classification :** Non-numeric data or nominal data decision trees 8. **Unsupervised learning and clustering:** Criterion functions for clustering, algorithms for clustering: K-means, Hierarchical and other methods, Cluster validation 9. **Applications:** Biometrics recognition, handwriting recognition, document recognition, multimedia data retrieval, speech recognition, data mining, web searching, network traffic analysis etc.
Suggested Books	1. R. O. Duda, P. E. Hart and D. G. Stork, **Pattern Classification**, John

	Wiley, 2001 2. S. Theodoridis and K. Koutroumbas, **Pattern Recognition**, 4th Ed., Academic Press, 2009 3. C. M. Bishop, **Pattern Recognition and Machine Learning**, Springer, 2006

Course Code	**CS 413**
Title of the Course	**Topics in Artificial Intelligence Programming**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Department of Computer Science & Engineering
Pre-Requisite, if any	Abstraction and Paradigms for Programming and Artificial Intelligence
Scope of the course	
Course Syllabus	**Basics of LISP and PROLOG** **AI Programming techniques**: Heuristic search and efficiency issues in search programs, Min-Max algorithm, Branch and Bound algorithm **Natural language parsing:** symbolic programming, lexical closures, memorization, **Object-oriented representations:** Common Lisp Object System (CLOS), hash tables, functions as first-class objects, macros, structures and lists **Rule-based expert systems:** Expert system with Prolog **Artificial neural networks:** Unsupervised Neural Networks, Destructive Operations, Automated Memorization, Supervised Neural Networks, Reinforce Learning **Game Playing:** Tournament
Suggested Books	1. P. Norvig, **Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp**, Morgan Kaufmann, 2000. 2. I. Bratko, **Prolog Programming for Artificial Intelligence** (3rd Ed), Pearson Education, 2001.

Course Code	**CS 414/ CS 614**
Title of the Course	**Cloud Computing and Applications**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science & Engineering
Pre-Requisite, if any	UG level courses on Operating Systems, Computer Architecture and Computer Networks
Scope of the Course	To study the technology behind the cloud computing methodology. The course would include many cloud computing service models namely *IaaS*, *SaaS*, and *PaaS* and cloud computing deployment models such as public Cloud, private Cloud and hybrid Cloud. Further, with the exponential growth in Cloud computing services, there is a need to understand the various issues that affect the different stakeholders of Cloud computing. The success story of the cloud computing not only depends on the underlying technology but also on the economics of the Cloud computing resource market. Hence, in this course, we would also cover the concept of Service Level Agreement (SLA), SLA matching techniques, cloud resources management, resource provisioning and sharing, pricing strategies, monitoring risk, trust, and Quality of Service (QoS) etc.
Course Syllabus	**History of Cloud Computing:** Paradigms in Computing, Parallel Computing, Distributed Computing, Grid Computing, Service Computing; Service Oriented Architecture (SOA), Web Services **Cloud Computing:** Definition, Characteristics, Architecture, Components, Service Models, Deployment Models, Virtualization: Server, Storage, Network, Desktop; Hypervisor, Virtual Machine, Multi-tenancy, Opportunities and Risks **Service Level Agreement (SLA):** Definition, Types of SLA, SLA Life Cycle, Issues Related to Cloud SLA, SLA Frameworks: WS-Agreement, WSLA, WSOL, Slang, Bilateral Protocol; Translation of SLAs into Monitoring Specifications, Dynamic Creation of Monitoring Infrastructures, Penalty Management, Runtime Prediction **Cloud Security:** Cloud Security Fundamentals, Vulnerability Assessment, Security and Privacy in Cloud, Cloud Computing Security Architecture: Identity Management and Access Control, Autonomic Security; VM Specific Security Techniques **Cloud Application Programming Models:** Cloud File Systems: GFS and HDFS, BigTable, HBase and Dynamo; Map Reduce Programming Model, Hadoop: Hadoop Fundamentals, Hama and other Hadoop Related Services

	Cloud Application Development Platforms: Xen Hypervisor, Amazon Web Service, Windows Azure, Google App Engine, Eucalyptus, Open Stack, Open Nebula
Suggested Books	1. A. T. Velte, **Cloud Computing - A Practical Approach**, McGraw Hills 2. P. Wieder and J.M. Butler, **Service Level Agreements for Cloud Computing**, Springer 3. C. Buan, **Cloud Computing - Web Based Dynamic IT Services**, Springer 4. Tanenbaum and V. Steen, **Distributed Systems: Principles and Paradigms**, Pearson 5. David E.Y. Sarna, **Implementing and Developing Cloud Computing Applications**, CRC Press 6. R. Krutz and R. D. Vines, **Cloud Security**, Wiley-India 7. T. White, **Hadoop: The Definitive Guide**, O'Reilly Media

Course Code	**CS 416/ CS 616**
Title of the Course	**Service Oriented Systems**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	UG Level course on Software Engineering and Computer Networks
Scope of the course	To understand the technical as well as management aspects of service-oriented systems. Emphasis would be on the most common realization of service-oriented systems i.e. web-services
Course Syllabus	• **Introduction:** service explosion in the world, independent services, 'servitization' of products • **Service-oriented systems:** understanding the 'register, find, bind' triangle, loose coupling, Software-as-a-Service, Governance issues Practical realization of service-oriented systems via web services, basics of xml and its use in web-service implementation, http protocol, utility of web-services • **Basic web services stack:** understanding the SOAP protocol, WSDL, UDDI registry. Implementation of web services using the basic web services stack • **Representational State Transfer (REST) web services:** implementation of RESTful web services, REST constraints, comparison of this approach of web-service implementation with that of the basic web-service stack, advantages and limitation of RESTful web services • **Service composition:** understanding of the concepts of service orchestration and service choreography, static versus dynamic service composition, assessment of quality in service compositions, appropriate service selection for compositions, role of the customer in service composition
Suggested Books	1. J. Snell, D. Tidwell, P. Kulchenko. **Programming Web Services with SOAP,** O'Reilly 2. L. Richardson, S. Ruby, D. H. Hansson. **Restful Web Services,** O'Reilly 3. B. A. Christudas, M. Barai, V. Cacello. **Service-Oriented Architecture with Java,** Packt Publishing.

Course Code	**CS 417/ CS 617**
Title of the Course	**Cryptography and Network Security**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department/Discipline	Computer Science and Engineering
Pre–requisite, if any	Discrete Mathematical Structures, Design and Analysis of Algorithms, Computer Networks
Scope of the course	To understand the basic concepts of cryptography, get familiarized with encryption and authentication protocols and look at system level security. We will study block ciphers, stream ciphers, hash functions and public key cryptography and security mechanisms in networks and Internet. In the process we will learn some number theory and algebra.
Course Syllabus	**Introduction:** What is cryptography, classical ciphers, cryptanalysis. **Shannon's theory:** Concept of perfect secrecy, entropy **Symmetric-key Cryptography:** Pseudorandomness, Stream ciphers, Block ciphers, Data Encryption Standards, Advanced Encryption Standards, Modes of operation **Hash-functions:** Data Integrity, Merkle-Damgard construction, Message Authentication Codes **Number Theory:** Euclidean Algorithm, Chinese Remainder Theorem, Primality Testing algorithms, Factoring algorithms **Public-key Cryptography:** RSA, Discrete log problem, Diffie-Hellman key exchange protocol, Signatures schemes Public key Infrastructure, Digital certificates **Network Security:** Network security at application,Security issues in electronic mail, IP Security, Web security, transport layer security and Secure Socket Layer, intrusion detection, malicious software, viruses, worms and related threats, firewalls, trusted systems.
Suggested Books	**Suggested Textbook:** 1. D. R. Stinson: **Cryptography theory and practices,** 3rd Edition, CRC Press, (2006) 2. W. Stalling: **Cryptography and Network security Principles and Practices,** 4th or 5th Edition PHI, 2006/2010 **Other References:** 1. Menezes, P. Oorschot, S. Vanstone: **Handbook of Applied**

	Cryptography (individual chapters are freely available online at http://www.cacr.math.uwaterloo.ca/hac/) 2. J. Katz and Y. Lindell: **Introduction to Modern Cryptography.** Chapman & Hall/CRC 2008 3. S. Singh: **The Code Book.** (A good popular introduction to the subject) Other web resources will be posted on the course website from time to time.

Course Code	**CS 418/ CS 618**
Title of the Course	**Systems and Usable Security**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	UG Level Courses on Operating Systems and Computer Networks
Scope of the course	To understand the principles of systems security from an applied viewpoint and obtain hands-on experience on security threats and counter-measures. To study operating systems security, advanced topics on network security, access control and digital rights management, web security and usable security. After the completion of the course, the student will have sound understanding of practical aspects of security and will be able to analyze and design the secure systems.
Course Syllabus	**Introduction:** Computer Security Concepts, threats, Attacks, and Assets **Malicious Software:** Types of Malicious Software (Malware), Infected Content–Viruses, Vulnerability Exploit–Worms, Social Engineering–SPAM E-mail, Trojans, System Corruption, Zombie, Bots, Information Theft–Keyloggers, Phishing, Spyware, Stealthing–Backdoors, Rootkits. **Operating System Security:** System Security Planning, Application Security, Linux/Unix Security, Windows Security, Virtualization Security **Access Control:** Access Control Principles, Subjects, Objects, and Access Rights, UNIX File Access Control, Role-Based Access Control, Attribute based Access Control. **Database Security:** The Need for Database Security, Database Management Systems, Database Access Control, Statistical Databases, Private Information Retrieval, Cloud Security. Digital Rights Management: Multicast security, copyright protection, Digital Finger printing. **Web Security:** Secure E-mail and S/MIME, Domain Keys Identified Mail, Secure Sockets Layer (SSL) and Transport Layer Security (TLS), HTTPS, IPv4 and IPv6 Security, Internet Authentication Applications, Kerberos, X.509, Public-Key Infrastructure, Federated Identity Management. **Wireless Security:** Wireless Security Overview, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security.

	Usable Security: Introduction to privacy, trust and semantic security, Visualizing privacy, Web browser security and privacy, Authentication and text passwords, biometrics and graphical passwords.
Suggested Books	1. W. Stallings and L. Brown, **Computer Security: Principles and Practice** (2nd Edition)**,** Prentice Hall, 2011. 2. A. Menezes, P. Oorschot, S. Vanstone: **Handbook of Applied Cryptography (**individual chapters are freely available online at http://www.cacr.math.uwaterloo.ca/hac/) **Other References:** 3. Goodrich and Tamassia, **Introduction to Computer Security,** Addison-Wesley, 2010. 4. Kaufman, Perlman and Speciner, **Network Security: Private Communications in a Public World***, (2nd edition),* Prentice Hall, 2003.

Course Code	**CS 419/ ICS 419/ CS 619**
Title of the Course	**Computer Vision**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	
Scope of the course	Objective of this course is to understand and create artificial vision systems which can reliably extract information from images. Study of vision problems require the basic understanding of image formation, image representation, ways of analyzing the images and patterns present in them. This course aims at providing the knowledge at all these fronts.
Course Syllabus	**Digital Image Processing:** Fundamentals, Types of Image Processing, Image Acquisition Methods, Human Perception of Color and Images, Transformations: Orthogonal, Euclidean, Affine, Projective etc. **Low-level Image Processing:** Image Enhancement in Spatial Domain – Histogram Processing, Contrast Stretching, Log Transformation, Gamma Correction, Smoothing and Sharpening; Logical and Arithmetic Operations, Morphological Image Processing, Image Enhancement in Frequency Domain, Fourier Transform, Convolution and Filtering, Image Restoration. **Image Feature Extraction:** Edge detection – Canny, Sobel, Prewitt, LOG, DOG, Line detector: Hough Transform; Corner detectors – Harris and Hessian Affine; Orientation Histogram, SIFT, SURF, HOG, GLOH, Scale-Space Analysis – Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT. **Image Segmentation:** Edge Based Approaches to Segmentation, Region Growing, Texture Segmentation, Object Detection and Segmentation: Graph-cuts, Active Contours, Mean-Shift. **Object Recognition:** Structural Approaches, Model-based Approaches, Appearance and Shape-based Approaches, Probabilistic Paradigms. **Pattern Analysis:** Clustering: K-Means; Gaussian Mixture Model (GMM); Classification – Discriminant Function, Supervised, Semi-supervised, Unsupervised; Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA; Non-parametric methods. **Motion Analysis:** Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis. **Applications and Performance Measures:** CBIR, CBVR, Activity Recognition, Biometrics, Document processing, Super-resolution, Augmented Reality, Security and Surveillance, Performance Evaluation Measures.

Suggested Books	**Text Books** 1. Computer Vision: A Modern Approach, D. A. Forsyth and J. Ponce, Pearson Education, 2003. (693 pages), ISBN: 9780130851987. 2. Computer Vision: Algorithms and Applications, Richard Szeliski, Springer-Verlag, 2011. (832 pages), ISBN: 978-1848829343. **Reference Books** 1. Digital Image Processing, Rafael C. Gonzalez and Richard E. Woods, Pearson Education, 2008. (976 Pages), ISBN: 9788131726952. 2. Pattern Classification, R.O. Duda, P.E. Hart and D.G. Stork, Wiley-Interscience, 2000. (654 pages), ISBN: 978-0471056690. 3. Multiple View Geometry in Computer Vision, Richard Hartley and Andrew Zisserman, Cambridge University Press, 2004. (668 pages), ISBN: 978-0521540513. 4. Introduction to Statistical Pattern Recognition, Keinosuke Fukunaga, Academic Press, 1990. (592 pages), ISBN: 978-0122698514.

Course Code	**CS 420/ CS 620**
Title of the Course	**Embedded Systems**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	Programming knowledge, Computer Architecture, Operating Systems. CSE students take these subjects in their I, II and III years.
Scope of the course	Embedded systems are becoming more and more ubiquitous and pervasive in our day to day life. Their applications range from domestic household appliances, health-care, defense, transportation, space technology, gaming, infotainment, mobiles, energy generation, etc. Research reports1 reveal that 99% of the microprocessors/software produced annually end up in embedded systems/applications. The course will focus on software issues in embedded systems. It will include demonstrations and getting acquainted with 8/16/32-bit micro-controller and its development environment (starter kits, appropriate compilers and flashers), interrupt programming, resource management, and peripheral interfacing & drivers. The practical part will involve demos and getting started kind of exercises to show the tangible side of taught concepts.
Course Syllabus	1. Introduction to embedded systems: Embedded vs. General purpose computer Systems; Abstract Model; computer-plant interaction and real-time reactive behaviour of embedded control systems. 2. Sequential and continuous control systems; 3. Basic modeling and implementation techniques for sequential and continuous control systems - state machines, function blocks and function block diagrams, which is followed by advanced modelling techniques for complex systems, such as hierarchical and concurrent state machines and hybrid models; 4. Real-time operating systems (RTOS), Real-time kernels, Deploying applications on RTOS/Kernels.
Suggested Books	1. David E. Simon, **Embedded Systems Primer**, Addison-Wesley, 1999, 020161569X / 9780201615692. 2. Tammy Noergaard, **Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers**, 2005, Newnes, ISBN-10: 0750677929, ISBN-13: 978-0750677929 / 9780123821966

Course Code	**CS 422/ CS 622**
Title of the Course	**Numerical Simulation**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science & Engineering
Pre–requisite, if any (for the students)	Calculus, Linear Algebra and Ordinary Differential Equations, Complex Analysis and Differential Equations, Numerical Methods
Scope of the course	Simulation is a useful tool in almost all areas of engineering and science. This course will introduce computational techniques for simulating applications from Electrical Engineering, Mechanical Engineering, Material Science, Physics, and Operations Research.
Course Syllabus	**Problem Types:** Circuit analysis, structural analysis of automobiles, analyzing drag force in aircrafts, engine thermal analysis, virtual environments for computer games, stock option pricing, electrostatic optimization for biomolecules etc. **Equation Formulation Methods:** Stamping, node-branch, and nodal. **Direct and Iterative Matrix Solution:** Error analysis, dense and sparse matrix factorizations, and Krylov methods. **Nonlinear Systems Solution:** Multi-dimension Newton, fixed-point and functional iterations, and continuation schemes. **Numerical Integration and Monte Carlo Methods:** Newton-Cotes, composite quadrature, Gauss quadrature, multiple integrals, generating samples, random tours, designing and analyzing random paths. **Discretization Methods for Partial Differential Equations:** Finite difference, finite element, multi-grid, and spectral methods.
Suggested books	1. G. Strang, **Computational Science and Engineering**, Wellesley-Cambridge Press (2007). 2. D. Kincaid and W. Cheney, **Numerical Methods: Mathematics of Scientific Computing**, Brooks / Cole (2002). 3. Y. Saad, **Iterative Methods for Sparse Linear Systems**, SIAM (2003). 4. C. T. Kelley, **Solving Nonlinear Equations with Newton's Method**, SIAM (2003). 5. E. L. Allgower and K. Georg, **Introduction to Numerical Continuation Methods**, SIAM (2003). 6. G. S. Fishman, **Monte Carlo Concepts, Algorithms, and**

	Applications, Springer (1996). 7. W. L. Briggs, V. E. Henson, and S. F. McCormick, **A Multigrid Tutorial**, SIAM (2000).

Course Code	**CS 424**
Title of the Course	**Functional and Logic Programming**
Credit Structure	L-T- P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre–requisite, if any	Computer Programming
Scope of the course	Functional-logic programming integrates most of the features of the classical declarative paradigms, namely of functional programming and of logic programming. From the functional paradigm it inherits named or anonymous functions, nested expressions, efficient reduction strategies, higher-order functions, and types. From the logic paradigm it inherits named relations, logical variables, partial data structures, unification, and built-in search. Applications include Semantic Web programs over metadata as well as Web Services that provide logic inferences and/or functional transformations over XML data.
Course Syllabus	**Introduction to Functional and Logic Programming:** Overview of Declarative Programming, Basic Notions of Functional Programming, Basic Notions of Logic Programming, Basic Notions of Functional-Logic Programming. **Terms in Functional and Logic Programming:** Taxonomy of Terms, Simple Terms, Complex Terms, Term Unification. **Functional and Logic Definition Clauses:** Taxonomy and Syntax of Clauses, Logic Clauses, Functional Clauses, Functional-Logic Clauses. **Higher Order Operations:** Function Composition, Compose as a higher order function, Relational Product as a higher order function. **Case Study:** Study of a Functional-Logic Programming Language (e.g. Relational Functional Markup Language (RFML))
Suggested books	1. J. Kelly, **The Essence of Logic**, , Prentice-Hall of India, 1997 2. H.B. Enderton, **Mathematical Introduction to Logic**, Academic Press, Elsevier, 2001 3. R. Wilensky, **Common LISPcraft**, W. W. Norton & Co., 1986 4. W.F. Clocksin and C.S. Melish, **Programming in Prolog**, Springer-Verlag, 5th Edition, 2004 5. G. Cousineau and M. Mauny, **The Functional Approach to Programming**, Cambridge University Press, 1998

Course Code	**CS 425 / CS 625**
Title of the Course	**Natural Language Processing**
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science & Engineering
Pre-requisite, if any	NA
Scope of the Course	The course is an introductory course in the natural language processing field. This is meant to get students familiar with the text processing techniques as well as more advanced techniques for text processing such as question answering, text summarization, parsing, semantic role labelling, etc.
Course Syllabus	Introduction NLP tasks in syntax, semantics, and pragmatics. Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. Brief history of the field. N-gram Language Models The role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models. Part Of Speech Tagging and Sequence Labeling Lexical syntax. Hidden Markov Models (Forward and Viterbi algorithms and EM training). Basic Neural Networks Any basic introduction to perceptron and backpropagation LSTM Recurrent Neural Networks "Understanding LSTM Networks" optionally the original paper Long Short Term Memory. Syntactic parsing Grammar formalisms and treebanks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs. Neural shift-reduce dependency parsing Semantic Analysis Lexical semantics and word-sense disambiguation. Compositional semantics. Semantic Role Labeling and Semantic Parsing. Information Extraction (IE) Named entity recognition and relation extraction. IE using sequence labeling. Machine Translation (MT) Basic issues in MT. Statistical translation, word alignment, phrase-based translation, and synchronous grammars.
Suggested Books	• D. Jurafsky & J. H. Martin, ***Speech and Language Processing,*** Pearson Education, India : India : 2013 : 9789332518414 • Manning and Schutze, ***Statistical Natural Language Processing***, MIT Press : Cambridge, MA : 1999 : 0262133601 • J. Allen, ***Natural Language Understanding,*** The Benajmins/ Cummings Publishing Company Inc. :1994 : 0-8053-0334-0 • Y. Goldberg and G. Hirst, ***Neural Network Methods in Natural Language Processing,*** Morgan & Claypool Publishers : 2017.: 978-1627052986

Course code	**CS 426 / CS 626**
Title of the course	**Foundations of Cyber-Physical Systems**
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Suitable for UG final/ Grad/ PhD students who studied courses of Discrete Mathematics, Automata Theory, Programming, Operating Systems
Scope of the course	This course aims to introduce the basics of Cyber Physical Systems, distinguishing characteristics that sets them apart from their other embedded system incarnations. The essential topics covered would be: Pnueli's Cactus model, real-time Vs. hybrid Vs. reactive systems, formal methods for rigorous analysis of CPS, concurrency, distributed algos (centralised as well as decentralised variants), Disentanglement of their complex nuances with decentralisation. The emphasis would be from systems specification, design and formal analysis perspectives.
Course Syllabus	1. Dynamical systems: Continuous Vs Discrete behaviour, Hybrid behaviour, Reactive Systems, CPS key features, Synchronous Vs. Asynchronous paradigms. 2. Control routine patterns: Different control cycle actuations, Event/ Time - triggered, Static cyclic scheduling. 3. Scheduling: Realtime Scheduling policies, Rate-monotonic, Deadline-monotonic, Utilisation-based Schedulability. 4. Kernels: Real-time kernels and Kernel-based system development. 5. Specification languages: State machines (deterministic, nondeterministic), structural vs behavioural specification 6. Correctness Analysis: Requirements specification via Live, Safe and Precedence properties, Timed Processes & Protocols, Timed Automata, Hybrid Automata (Linear vs Non-linear), Mechanised Proof techniques, Deductive verification, 7. CPS Designs: From various application domains and CPS Case Studies.
Suggested Books	1. A. Platzer, "***Logical Foundations of Cyber-Physical Systems***", Springer, Switzerland, ISBN 978-1-4419-8236-0, 2017. 2. E. A. Lee and S. A. Seshia, "***Introduction to Embedded Systems: A Cyber-Physical Systems Approach***", Second Edition, MIT Press, Cambridge (MA, USA), ISBN 978- 0-262-53381-2, 2017. 3. C. Baier and J. P. Katoen, "***Principles of Model Checking***", MIT Press, London (UK), ISBN: 9780262026499, 2008.

Course code	**CS 427/ CS 627**
Title of the course	Advanced Computer Networks
Credit Structure	L-T-P–Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Computer Networks (UG Level Course) Operating Systems (UG Level Course)
Scope of the Course	This course will give a background on design principles of highperformance networking devices like switches and routers. It will introduce advanced topics and recent trends in Computer Networks like Software Defined Networking, Data Center Networks, Information Centric Networking and Future Internet. The Learning outcomes are: 1. Introduce PG/UG students to the recent advances and state-of-theart Computer Networking topics (Architecture/Protocol/Systems), and also develop understandings on the future research aspects in networking to help guide the students towards potential MTech/MS/PhD work. 2. Provide insights on the principles and design decisions behind networking aspects, Internet technologies and future research. 3. Expose students to hands-on exercises, work on network simulators and help them design and implement the networking protocols and applications.
Course Syllabus	High Performance Switching and Routing: 1. IP Address Lookup- Trie-based Algorithms, Hardware Lookup. 2. Quality of Service: Need for Packet Classification, Different Classification Methods, TCAM based Classification, Differentiated Service, Traffic Polishing, Traffic Shaping, Packet Scheduling, Queue Management Techniques. 3. Packet Switching: Switching Overview, Switching Fabric, Buffering in Switch Fabric, Multiplane and Multistage Switching Network Softwarization: 1. Software Defined Networking (SDN): Control and Data Plane Separation. Network Operating System (SDN Controllers), Intent based Networking (IBN), Southbound Interface (OpenFlow) and OpenVSwitch. 2. Programmable Networking Devices: P4, Smart Switches, and SmartNICs. 3. Network Virtualization: VirtIO, SR-IOV, and Network Function Virtualization (NFV). Data Centre Networking: 1. Networking Topologies: Fat-Tree, Clos, Leaf-Spine, Docker and Container Networking Interface (CNIs), Kubernetes. Switching in Data Centre Networking, Virtual Switches.

	2. High-Performance Computing Networks: HPC System Architectures, HPC Networking Standards, HPC Networking Software, Low Latency Ethernet (10 GbE, 40 GbE, 100 GbE, InfiniBand FDR). Named Data Networking: Content Distribution on the Internet, Architectures for Information Centric Networking, Content Naming, Routing and Caching, Security in Named Data Networking.
Suggested Books	1. High Performance Switches and Routers, H. Jonathan Chao, Bin Liu, 2007, John Wiley & Sons, Inc. ISBN-10: 0-470-05367-4 2. Information-Centric Networks: A New Paradigm for the Internet (Focus Series in Networks and Telecommunications), Gabriel M. de Brito, Pedro B. Velloso, Igor M. Moraes, Wiley-ISTE; 1st edition, 2013, ISBN: 9781848214491 3. Information-Centric Networking (ICN): Content Centric Networking (CCNx) and Named Data Networking (NDN) Terminology, B. Wissingh, C. Wood, A. Afanasyev, L. Zhang, D. Oran and C. Tschudin, RFC 8793, June 2020 (All RFCs are free documents) 4. Software-Defined Networks: A Systems Approach, Peterson, Cascone, O'Connor, Vachuska, and Davie, Online Free Reference Book (https://sdn.systemsapproach.org/index.html) 5. Cloud Networking: Understanding Cloud-based Data Centre Networks, Gary Lee (Author), Morgan Kaufmann (Publisher), 2014, ISBN-139780128007280 6. Relevant Request for Comments (RFC) - IETF http://www.ietf.org/rfc.html 7. Research Publications – relevant works will be discussed and distributed time to time

Course code	**CS 428/ CS 628**
Title of the course	**Algorithmic Graph Theory**
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science & Engineering
Pre-requisite, if any	Discrete Mathematical Structures, Data Structures and Algorithms.
Scope of the course	This course shall impart basic background on the theoretical concepts of graph theory. The topics covered shall cover basic concepts and algorithmic aspects, such as graph representations, terminologies, and properties of different types of graphs, connectivity properties and algorithms, as well as some advanced topics, such as graph coloring, matching, planarity, and spectral graph theory. Undergraduate and postgraduate students shall get firm foundations in solving real-world problems such as path optimization problems, and other graph-theoretical problems that are relevant to theoretical computer science and operations research.
Course Syllabus	• **Introduction to graphs, trees, and their properties:** Graphs, Representation of Graphs, Various Special Graphs, Walks, Graph Isomorphism, Spanning Trees, Counting Spanning trees in polynomial time, Algorithms for minimum weighted spanning trees. • **Matching Algorithms and Cycles in Graphs:** Matching, Perfect matching, Augmenting path algorithm, Bipartite matching algorithm, Hall Marriage Theorem, Konig's theorem, Matching in general graphs, Tutte's Theorem, Eulerian tour and Seven Bridges problem, Hamiltonian cycles and Travelling Salesman Problem, Necessary Conditions for Hamiltonian Graphs, Sufficient Conditions for Hamiltonian Graphs. • **Coloring and Connectivity in graphs:** Vertex Coloring, Edge Coloring, Brook's theorem, Vizing Conjecture. Vertex and Edge Connectivity, Vertex- and edge-disjoint paths, testing connectivity, Algorithm for the cut-vertices, Algorithm for decomposing connected graph into blocks, Tutte's decomposition, edge-connectivity, Menger's Theorem. • **Network Flow Algorithms:** Basic concepts on flows and networks, max-flow min-cut theorem, Ford-Fulkerson algorithm. • **Planarity in graphs:** Planar graphs, Euler's Formula, Outer Planar Graphs, Kuratowski Theorem, Four Color Theorem. • **Spectral graph theory:** Adjacency matrix, Laplacian matrix, Random regular graphs, Expander graphs, Ramanujan graphs. • **Applications and Case studies:** Social Network Analysis, Complex Network Analysis.

Suggested Books	• D. B. West: Introduction to Graph Theory: Pearson Education: India : 2015: 8178088304. • R. Diestel: Graph Theory: Springer-Verlag: New York: 2000: 0387950141. • R.B. Bapat: Graphs and matrices: Springer. : London: 2010: 9789380250694. • Bondy and U. S. R. Murthy: Graph Theory, Graduate Texts In Mathematics: Springer : Switzerland: 2008: 978-1-84628-969-9. • Alan Gibbons : Algorithmic Graph Theory: Cambridge University Press: 1985: 9780521288811. • T. Cormen and C.E. Leiserson and R.L. Rivest and C. Stein: Introduction to Algorithms: The MIT Press: Third Edition, Sept 2009, 9780-262-03384-8 • Narsingh Deo: Graph Theory with Applications to Engineering and Computer Science: PHI Learning: 9788120301450

Course code	**CS 630/ CS 430**
Title of the course	**Data Center Networking**
Credit Structure	L-T-P–Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Students must have knowledge of UG-level computer network courses and have good programming skills in python and C/C++.
Scope of the Course	In this course, students are expected to learn the data center network architectures, their underlying protocols and understand the challenges faced in designing a data center.
Course Syllabus	• Introduction to network architectures—Edge, Metro and Core Networks Evolution of Data Center Networks—a switch-centric DCN, a modular DCN, a wireless DCN, and a hybrid DCN. • Datacenter architectures and their building blocks—Network fabric, cross connect design, Server addressing and routing protocols. Multipath routing. Overview of Software Defined Networking and Network Function Virtualization. • State-of-the-art of data center networking architectures—Fat tree, Helios, VL2, Portland, B-cube, Wavecube etc. • Datacenter telemetry and resource management—traffic engineering, congestion control, load balancing and resource allocation algorithms. Datacenter network performance enhancement strategies. • Challenges involved in building a datacenter—cross-connect and protocol scalability, fault tolerance, security, power etc. Recent advancements in data centers—hybrid datacenter architectures etc..
Suggested Books	**Text Book:** 1) Yang Liu, Jogesh K. Muppala, Malathi Veeraraghavan, Dong Lin, Mounir Hamdi, "Data Center Networks: Topologies, Architectures and Fault-Tolerance Characteristics," Springer, 2013: ISBN-9783319019482. **Reference books:** 1) Guo, D. **"Data center networking: Network topologies and traffic management in large-scale data centers,"** Singapore, Springer, 2022: ISBN—9789811693687. 2) M. Arregoces and M. Portolani, **"Data Center Fundamentals,"** Cisco Press, 2004: ISBN—1587050234. 3) L. Zhang and L. Chen, **"Cloud Data Center Network Architectures and Technologies,"** CRC Press, 2021: ISBN—9780367695705

	4) J. Donovan and K. Prabhu, **"Building the Network of the Future: Getting Smarter, Faster, and More Flexible with a Software Centric Approach (100 Cases),"** CRC Press, 2017: ISBN—9781138631526

Course Code	**CS 632/ CS 432**
Title of the Course	**Reinforcement Learning**
Credit Structure	L-T-P-Credit 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Students should have good understanding of linear algebra, probability and statistics, knowledge of algorithm design and analysis, and proficiency in programming.
Scope of the Course	• Learning the basic theory of reinforcement learning. • Understanding a range of reinforcement learning algorithms with their strengths and limitations. • Formulation of reinforcement learning problems for different applications.
Course Syllabus	**Introduction:** Types of machine learning approaches: supervised, unsupervised, reinforcement; Elements of reinforcement learning: agent, environment, policy, reward and value functions **Multi-armed bandits:** Sequential decision making: exploration and exploitation; bandit problem solution: greedy, optimistic-greedy, and epsilon-greedy algorithms; Upper confidence bound bandit algorithm. **Markov Processes:** Markov property, Markov chains, Markov reward process, Markov decision process (MDP), Bellman expectation equation, optimal value function, optimal policy, Bellman optimality equation. **Dynamic Programming:** Dynamic programming for MDP, iterative policy evaluation, policy improvement, policy and value iteration. **Monte Carlo Methods:** Model free reinforcement learning, Monte Carlo policy evaluation and estimation of action values, on- and off-policy **Temporal-Difference Learning:** Temporal-Difference (TD) prediction, TD(0), TD(1), TD(λ), TD control methods: SARSA, Q-Learning and variants. **Function Approximation Methods:** Risk minimization, eligibility trace for function approximation, value function approximation (VFA), Monte Carlo learning and TD learning for policy evaluation with linear VFA. **Policy Gradients:** Policy gradient methods, policy search methods, gradient-free methods, finite difference methods, likelihood ratio policy gradient, bias and variance in reinforcement learning, actor-critic methods.
Suggested Books:	**Text Book:** 1. R. S. Sutton and A. G. Barto, **"Reinforcement Learning - An Introduction"**, MIT Press, Cambridge, USA, 1998, ISBN-13: 978-0262193986 **Reference Books:** 2. P. Winder, **"Reinforcement Learning: Industrial Applications of Intelligent Agents**", O'Reilly Media, Inc, USA, 2020, ISBN-13:978-1098114831 3. K. P. Murphy, "**Machine Learning: A Probabilistic Perspective**", The MIT Press, Cambridge, USA, 2012, ISBN-13: 978-0262018029

	4. I. Gridin, "**Practical Deep Reinforcement Learning with Python**", BPB Publication, India, 2022, ISBN-13: 978-9355512055.

Course Code	**CS 334/ CS 434/ CS 634**
Title of the Course	**Wireless Networks and Applications**
Credit Structure	L-T-P–Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Students must have knowledge of UG-level computer network courses and have good programming skills in python and C/C++.
Scope of the Course	This course covers a broad range of wireless networking standards including 5G/6G, and reviews important wireless network application areas. This course will provide a basic understanding and working of wireless networks to the students.
Course Syllabus	• Introduction to wireless networks. An overview of layered architecture, addressing and forwarding in LANs and IP networks from a wireless perspective. • IEEE 802.11—Physical Layer Standards, Diversity & Rake Receivers, Spread Spectrum, Multicarrier Modulation and Orthogonal Frequency Division Multiplexing (802.11a and 802.11g). Wireless channel characteristics. • Overview of wireless MAC protocols and management functions. Enhancements to support quality of service (802.11e). Overview of cellular standards GSM, GPRS, CDMA, LTE, 5G/6G and Mobility. • Wireless in today's Internet—TCP over wireless, IEEE 802.11 architecture, IAPP (Inter Access Point Protocol), LoRaWAN, 6LoWPAN. Overview of wireless Ad-hoc networks.
Suggested Books:	**Textbook:** **1.** C. Beard and W. Stallings, **"Wireless Communication Networks and Systems,"** Pearson, first edition, 2015, ISBN: 9780133594171. **Reference Books:** 2. D. Tse, P. Viswanath, **"Fundamentals of Wireless Communication,"** Cambridge University Press, 2005, ISBN: 0521845270. 3. Y. C. Eldar, A. Goldsmith, D. Gündüz, **"Machine Learning and Wireless Communications,"** Cambridge University Press, 2022, ISBN: 1108832989. 4. Savo Glisic, **"Advanced Wireless Networks: Technology and Business Models,"** Wiley, 2016: ISBN—9788126565016. **5.** J. Geier, **"Designing and Deploying 802.11 Wireless Networks: A Practical Guide to Implementing 802.11n and 802.11ac Wireless Networks For Enterprise-Based Applications (Networking Technology),"** Cisco press, 2015: ISBN—9781587144301.

Course Code	**CS 451** [From AY 2010-11 to 2013-14]
Title of the Course	**Soft Computing Lab**
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	Experiments to support the associated theory course that demonstrate the different applications of soft computing to Optimization; Functional approximation; Time-series prediction; Pattern recognition; Data compression; Control applications.
Suggested Books	Same as the associated theory course CS 401: Soft Computing

Course Code	**CS 452** [**CS 359** from AY 2015-16 onwards]
Title of the Course	**Parallel computing Lab** [From AY 2010-11 to 2013-14]
Credit Structure	L-T-P-Credits 0-0-3-1.5
Name of the Concerned Department	Computer Science and Engineering
Pre–requisite, if any	A course in Computer Programming
Scope of the course	
Course Syllabus	Experiments to support the associated theory course.
Suggested Books	Same as the associated theory course CS 402: Parallel Computing

Course Code	**CS 334/ CS 434/ CS 634**
Title of the Course	**Wireless Networks and Applications**
Credit Structure	L-T-P Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Students must have knowledge of UG-level computer network courses and have good programming skills in python and C/C++.
Scope of the Course	This course covers a broad range of wireless networking standards including 5G/6G, and reviews important wireless network application areas. This course will provide a basic understanding and working of wireless networks to the students.
Course outcome	The students will learn about the new generation network technologies.
Course Syllabus	● Introduction to wireless networks. An overview of layered architecture, addressing and forwarding in LANs and IP networks from a wireless perspective. ● IEEE 802.11 Physical Layer Standards, Diversity and Rake Receivers, Spread Spectrum, Multicarrier Modulation and Orthogonal Frequency Division Multiplexing (802.11a and 802.11g). Wireless channel characteristics. ● Overview of wireless MAC protocols and management functions. Enhancements to support quality of service (802.11e). Overview of cellular standards GSM, GPRS, CDMA, LTE, 5G/6G and Mobility. Wireless in today's Internet TCP over wireless, IEEE 802.11 architecture, IAPP (Inter Access Point Protocol), LoRaWAN, ● 6LoWPAN. Overview of wireless Ad-hoc networks.
Suggested Books:	Textbook: 1. C. Beard and W. Stallings, ***Wireless Communication Networks and Systems***, Pearson, first edition, 2015, ISBN: 9780133594171. Reference Books: 2. D. Tse, P. Viswanath, ***Fundamentals of Wireless Communication***, Cambridge University Press, 2005, ISBN: 0521845270. 3. Y. C. Eldar, A. Goldsmith, D. Gündüz, ***Machine Learning and Wireless Communications***, Cambridge University Press, 2022, ISBN: 1108832989. 4. Savo Glisic, ***Advanced Wireless Networks: Technology and Business Models***, Wiley, 2016. ISBN: 9788126565016. 5. J. Geier, ***Designing and Deploying 802.11 Wireless***

	Networks: A Practical Guide to Implement 802.11n and 802.11ac Wireless Networks For Enterprise-based Applications (Networking Technology), Cisco press, 2015. ISBN: 9781587144301.

Course Code	CS 401 / CS 601 (From AY 2024-25 onwards)
Title of the Course	**Soft Computing**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Prerequisite, if any	Discrete Mathematical Structures, Design and Analysis of Algorithms, Computational Intelligence
Scope of the Course	After having basic knowledge of artificial intelligence related to neural, fuzzy and evolutionary approaches, advancements in different areas are to be covered with working in a specific domain. This is by taking a case study to come up with the implementation and results.
Course outcome	The students will learn the theoretical and practical concepts of soft computing, hybrid intelligent systems, adaptation and applications of novel systems.
Course Syllabus	• **Review on Mathematical and theoretical methods on soft computing:** Neural networks. RBF structures. Self- organizing networks and methods. Fuzzy logic. Support vector machines and kernel methods. Evolutionary algorithms. • **Hybrid Intelligent Systems:** Neuro-fuzzy systems. Neuro-Genetic systems, Evolving neural systems. Neuro-swarm. Hybridization with novel computing paradigms: Quantum computing, DNA computing, membrane computing. Neural dynamic logic and other methods, etc. • **Learning and adaptation for novel systems:** Adaptive systems. Imitation learning. Reconfigurable systems. Supervised, unsupervised, Semi-supervised, reinforcement and statistical algorithms. Stability and convergence analysis. • **Applications:** Image and signal processing. Ambient intelligence. process control, and manufacturing. Biometry and bioinformatics. Data mining. Internet modeling, communication and networking. Intelligent systems in education. Human robot interaction. Time series analysis and prediction etc.
Suggested Books	Textbooks: 1. R. Jang and Mizutani, ***Neuro-Fuzzy and Softcomputing: A Computational Approach to Learning and machine Intelligence***, Pearson, 1996. ISBN: 978-0132610667 2. R. John and Ralph Birkenhead, ***SoftComputing Techniques and Applications (Advances in Intelligent and Softcomputing)***, Springer-Verlag, 2000. ISBN: 978-3790812572 Reference books: 3. F.O. Karray, C. W. De Silva, ***SoftComputing and Intelligent System Design: Theory, Tools and Applications***, Addison Wesley; 2009. ISBN: 978-8131723241.

Course code	CS 403/ CS 603
Title of the course	**Machine Learning**
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Artificial Intelligence/Computational Intelligence
Objective(s)	● This course provides a broad introduction to machine learning, datamining, and statistical pattern recognition. Topics include supervised learning, unsupervised learning, best practices in machine learning ● The course will also draw from numerous case studies and applications, so that the candidate's alos can learn how to apply learning algorithms to build different intelligent systems.
Course Outcome	The students will learn the basics of ML and its application.
Course Syllabus	● Introduction, Machine Learning and AI, Motivations for Studying ML, ● Supervised and Unsupervised learning, Linear prediction, Maximum likelihood Regularizers, basis functions and cross-validation, Optimisation, ● Linear and Logistic Regression, Gaussian Discriminant Analysis, Support Vector Machines, Decision Trees, Neural networks architectures and its advances, Ensemble Methods, Clustering, Naive Bayes, Bayesian Statistics, K-Means, Gaussian Mixture Models, Learning Theory, Model Selection.
Suggested Books	Texbooks: 1. C. M. Bishop, ***Pattern Recognition and Machine Learning***, Springer, Heidelberg, 2006. ISBN: 978-0-387-31073-2 2. T. Mitchell, ***Machine Learning***, McGraw Hill, 1997 (new chapters on line, 2006), New York, 1997. ISBN: 978-0071154673 Reference books: 3. Duda, Hart and Stork, ***Pattern Classification*** (2nd ed.), Wiley Interscience, US, 2000. ISBN: 978-8126511167

Course Code	**CS 406 / CS 606**
Title of the Course	**Data Mining and Data Warehousing**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-Requisite, if any	Basics of Data Base and Information Systems
Objective(s)	The course will cover the advanced concepts of data mining and warehousing
Course outcome	The students will learn the data warehousing and technologies, and data mining.
Course Syllabus	● **Data Warehouse and OLAP Technology:** Data warehousing Definition, usage and trends, Data marts, Metadata, Multidimensional data model, Data cubes, Schemas for Multidimensional Database: stars, snowflakes and fact constellations, Data warehouse architecture, OLTP and OLAP, types of OLAP servers: ROLAP, MOLAP, 3- Tier data warehouse architecture, Data warehouse implementation, computation of data cubes, indexing OLAP data, processing OLAP queries. ● **Data Mining:** Data mining definition and task, data preprocessing, data mining functionalities: Characterization and Discrimination, Mining frequent patterns, Frequent itemset mining methods, associations, and Correlations, Classification and Predictions, Cluster Analysis, Outlier Analysis, Evolution Analysis ● **Mining complex data objects:** Spatial databases, Multimedia databases, Time series and Sequence data, mining Text Databases and mining Word Wide Web, Applications and Trends in Data Mining
Suggested Books	Textbooks: 1. J. Han and M. Kamber, ***Data Mining: Concepts and Techniques***, Elsevier Publication, 2011. ISBN: 978-9380931913 Reference books: 2. M. H. Dunham, ***Data Mining: Introductory and Advanced Topics***, Pearson Education, 2006. ISBN: 978-8177587852

Course Code	**CS 407**
Title of the Course	**Peripherals and Interfaces**
Credit Structure	L-T- P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre requisite, if any	Computer Architecture
Objective(s)	This course deals with the various aspects of hardware software interfacing with peripherals and associated devices. The course covers the fundamentals of various peripheral devices, its programming through assembly language and architecture. Further, it provides the an avenue for learning concepts of microprocessors, microcontrollers, interrupts and memory access mechanisms.
Course outcome	The students will learn the concepts of ● interfacing techniques with hardwares and softwares ● designing the interface hardware
Course Syllabus	● Basics of Microprocessor: Design, Memory Subsystems, System Resources, Types and Interrupt handling, 8085 Architecture and its programming, 8086 ● Architecture and its programming, DMA channel, I/O port addresses. I/O buses, Local bus, DMA controller, PCI, ADC/DAC interfacing with microcontrollers/microprocessors. ● GPUs, USB, Bluetooth, 8255 interfacing, RAID. Video Hardware, Video display technologies, ● Introduction to serial communication, 8253/8254 programmable timer and interval counter. ● I/O Interfaces, USB Basic and Driver model Testing of serial and parallel port, USB mouse/keyboard interfaces. ● Interrupt Controller, Video/Graphics of Modern Desktop Board, Concepts of Network Interface Card, Design and Integration of Peripheral devices to a computer system as a Case Study.
Suggested books	Textbooks: 1. Douglas V. Hall. ***Microprocessor and Interfacing: Programming and Hardware***. McGraw Hill Inc., 1991. ISBN: 978-0070257429 2. Ramesh S. Gaonkar, ***Microprocessor Architecture, Programming and Application with the 8085***, Penram Int. Pub., 2013. ISBN: 978-8187972884 Reference books: 3. Stuart R. Ball. ***Analog Interfacing to Embedded Microprocessors***, Elsevier, 2003. ISBN: 9780080469973

Course Code	**CS 409 / CS 609**
Title of the Course	**Advanced Topics in Database Management Systems**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Department of Computer Science and Engineering
Pre-Requisite, if any	Data Structures and Algorithms and Database and Information Systems
Objective(s)	The course will cover the advanced topics on database management and security aspects.
Course outcome	The students will learn the advanced topics of data base management
Course Syllabus	• **Advanced Data Models:** Enhanced Relational System, Object-Oriented Data Model, Spatial and Temporal Databases, Multimedia Databases. • **Query Processing and Optimization:** Query Interpretation and Equivalence Expressions, Cost Estimate in Query Optimization, Semantic Query Optimization. • **Transaction Processing and Concurrency Control:** Properties of Transactions, Schedules and Serializability of Schedules, Transaction Failures and Recoverability, High Performance Transaction Systems. Distributed Databases: Design of Distributed Databases, Distributed Query Processing, Deadlock Handling, Concurrency Control and Recovery. • **Database Security and Authorization:** Database Security Issues, Security and Integrity Violations, Multilevel Security, Discretionary and Mandatory Access Control, Statistical Database Security.
Suggested Books	Textbooks: 1. R. Elmasri and S. Navathe, ***Fundamentals of Database Systems***, 7th Edition, Pearson, 2015. ISBN: 978-0133970777 2. H. F. Korth and A. Silberschatz, ***Database System Concepts***, McGraw Hill Inc., 2019. ISBN: 9780078022159 Reference books: 3. C. Zaniolo, S. Ceri, C. Faloutsos, R. T. Snodgrass, V.S. Subrahmanian, R. Zicari, ***Advanced Database Systems,*** Morgan Kauffmann, 1997. ISBN: 978-1558604438

Course Code	**CS 410**
Title of the Course	**Genetic Algorithms**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Department of Computer Science and Engineering
Pre-Requisite, if any	Optimization Algorithms and Techniques
Objective(s)	The course will cover the topics on genetic algorithms
Course outcome	The students will learn the concepts of genetic algorithms
Course Syllabus	● **Evolutionary Computations:** Biological background, Canonical GA ● framework, Basic Terminologies, Formulation of Optimization problems into GA framework. ● **Variations of GAs:** Binary Coded GAs and its variations such as Micro GA, Messy GA, Greedy GA etc., Real Coded GAs, Permutation Encoding GA etc. ● **GA operators:** Selection, Reproduction, Crossover, Mutation etc. Convergence criteria, Mathematical Construction of Genetic Operators, Schema Theorem of John Holland. ● **Advanced Operators and Techniques in GA:** Diploidy and Multiploidy, Inversion and Reordering, Niche and Speciation, Segregation and Translocation. ● **Multi-Objective GAs:** Non Pareto and Pareto-based GAs, MOGA, NSGA, Niched Pareto Genetic Algorithm. ● **Practice of GA with some real-life problems and GA Programming:** Traveling Salesman Problem, Word Matching problem, Topological Planning in Wireless Network, Placement and Routing problem in VLSI Design, Image Processing and Pattern Recognition.
Suggested Books	Textbooks: 1. D. E. Goldberg, Genetic Algorithms in Search , ***Optimization and Machine Learning***, Pearson Education, 2000. ISBN: 978-0201157673 2. K. Deb, ***Multi-Objective Optimization using Evolutionary Algorithms***, John-Wiley and Sons, Ltd. Chichester, 2010. ISBN: 978-8126528042 Reference books: 3. T. Back, David B. Fogel, Z. Michalewicz, ***Handbook of Evolutionary Computation***, Oxford University Press, 1999. ISBN: 978-0750308953 4. M. Mitchell, ***An Introduction to Genetic Algorithms*** (3rd Ed) Bradford Book, 1998. ISBN: 978-0262631853

Course code	**CS 411 / CS 611**
Title of the course	**Advanced Algorithms**
Course Category	Institute Elective
Credit Structure	2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Data Structures and Algorithms and Design and Analysis of Algorithms and TOC
Objective(s)	This course will introduce the advanced concepts of data structures and algorithm design
Course Outcomes	Students will advanced concepts of data structures and algorithm design
Course Syllabus	• **Advanced data Structures:** Binomial heaps and Fibonacci heaps, Red-Black tree, Splay tree, van Emde Boas Priority Queues. • **String Algorithms:** Rabin-Karp Fingerprinting Algorithm, KMP algorithm, Suffix Tries. • **Computational Geometry:** Convex Hull. Line-segment Intersection. Sweep Lines. Voronoi Diagrams • **Flow Algorithms:** Augmenting Paths and Push-Relabel Methods, Max-flow Min-cut Theorem, Minimum Cost Flows, Bipartite Matching. • **Linear Programming:** Linear Programming Duality, Interior Point method • **Complexity Theory and Approximation Algorithms**: Counting number of spanning trees vs. perfect matching, NPC, Approximation Algorithms
Suggested Books	Textbooks: 1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, ***Introduction to Algorithms***, (3rd Edition), Prentice Hall, 2009. ISBN: 978-81-203-4007-7 Reference books: 1. Ravindra Ahuja, Thomas Magnanti, and James Orlin, ***Network Flows: Theory, Algorithms, and Applications***, (Pearson), 1993, ISBN: 978-0136175490 2. Jon Kleinberg and Eva Tardos, ***Algorithm Design***, Pearson, 2005, ISBN 0-321-29535-8

Course Code	CS 412/ CS 612
Title of the Course	**Pattern Recognition**
Credit Structure	L-T-P-Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-Requisite, if any	Basics of probability theory, Programming
Objective(s)	This course aim to cover the basic concepts for analyzing patterns and their preprocessing techniques. It also aims to give exposure to various learning algorithms and their applications to various real life applications.
Course outcome	The students will learn basic concepts for analyzing patterns and their preprocessing techniques
Course Syllabus	• **Basics of pattern recognition:** Definitions, data sets for pattern recognition, representations of patterns and classes, metric and non-metric proximity measures, feature extraction, statistical and syntactic pattern recognition • **Bayesian decision theory:** Classifiers, discriminant functions, decision surfaces, normal density and discriminant functions, discrete features • **Parameter estimation methods:** Maximum-likelihood estimation, expectation-maximization method, Bayesian estimation, Gaussian mixture models • **Non-parametric techniques:** Density estimation using Parzen-window method, K-nearest neighbor method, nearest neighbor classifier • **Dimension reduction methods:** Lineardiscriminant analysis (LDA), principal component analysis (PCA) • **Linear discriminant function based classifiers:** Perceptron, support vector machines (SVM) • **Non-metric methods for pattern classification:** Non-numeric data or nominal data decision trees • **Unsupervised learning and clustering:** Criterion functions for clustering, algorithms for clustering: K-means, Hierarchical and other methods, Cluster validation • **Applications:** Biometrics recognition, handwriting recognition, document recognition, multimedia data retrieval, speech recognition, data mining, web searching, network traffic analysis etc.
Suggested Books	Textbooks: 1. R. O. Duda, P. E. Hart and D. G. Stork, ***Pattern Classification***, John Wiley, 2007. ISBN: 978-8126511167 2. S. Theodoridis and K. Koutroumbas, ***Pattern Recognition***, 4th Ed.,

	Academic Press, 2009. ISBN: 978-1597492720 Reference books: 3. C. M. Bishop, ***Pattern Recognition and Machine Learning***, Springer, 2016. ISBN: 978-1-4939-3843-8

Course Code	CS 414/ CS 614
Title of the Course	**Cloud Computing and Applications**
Credit Structure	L-T-P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-Requisite, if any	UG level courses on Operating Systems, Computer Architecture and Computer Networks
Scope of the Course	● To study the technology behind the cloud computing methodology. ● Further, with the exponential growth in Cloud computing services, there is a need to understand the various issues that affect the different stakeholders of Cloud computing.
Course outcome	The students will learn basic concepts cloud computing, cloud-based services, and SLA.
Course Syllabus	● **History of Cloud Computing:** Paradigms in Computing, Parallel Computing, Distributed Computing, Grid Computing, Service Computing; Service Oriented Architecture (SOA), Web Services ● **Cloud Computing:** Definition, Characteristics, Architecture, Components, Service Models, Deployment Models, ● **Virtualization:** Server, Storage, Network, Desktop; Hypervisor, Virtual Machine, Multi- tenancy, Opportunities and Risks ● **Service Level Agreement (SLA):** Definition, Types of SLA, SLA Life Cycle, Issues Related to Cloud SLA, SLA Frameworks: WS- Agreement, WSLA, WSOL, Slang, Bilateral Protocol; Translation of SLAs into Monitoring Specifications, Dynamic Creation of Monitoring Infrastructures, Penalty Management, Runtime Prediction ● **Cloud Security:** Cloud Security Fundamentals, Vulnerability Assessment, Security and Privacy in Cloud, Cloud Computing Security Architecture: Identity Management and Access Control, Autonomic Security; VM Specific Security Techniques ● **Cloud Application Programming Models:** Cloud File Systems: GFS and HDFS, BigTable, HBase and Dynamo; Map Reduce Programming Model, Hadoop: Hadoop Fundamentals, Hama and other Hadoop Related Services ● **Cloud Application Development Platforms:** Xen Hypervisor, Amazon Web Service, Windows Azure, Google App Engine, Eucalyptus, Open Stack, Open Nebula
Suggested Books	Textbooks: 1. A. T. Velte, ***Cloud Computing - A Practical Approach***, McGraw Hills, 2017. ISBN: 978-0070683518 2. P. Wieder and J.M. Butler, ***Service Level Agreements for Cloud Computing***, Springer, 2011. ISBN: 978-1461416135 Reference books: 3. C. Buan, ***Cloud Computing - Web Based Dynamic IT Services***, Springer, 2011. ISBN: 978-3642209161

	4. Tanenbaum and V. Steen, ***Distributed Systems: Principles and Paradigms***, Pearson, 2016. ISBN: 978-1530281756 5. David E.Y. Sarna, ***Implementing and Developing Cloud Computing Applications***, CRC Press, 2010. ISBN: 978-1439830826 6. R. Krutz and R. D. Vines, ***Cloud Security***, Wiley, 2010. ISBN: 978-0470589878 7. T. White, ***Hadoop: The Definitive Guide***, O'Reilly Media, 2009. ISBN: 9780596521974

Course Code	CS 416/ CS 616
Title of the Course	**Service Oriented Systems**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre requisite, if any	UG Level course on Software Engineering and Computer Networks
Objective(s)	● To understand the technical as well as management aspects of service-oriented systems. ● Emphasis would be on the most common realization of service-oriented systems i.e. web-services
Course outcome	The students will learn the basic functionalities of service oriented systems.
Course Syllabus	● **Introduction:** service explosion in the world, independent services, 'servitization' of products ● **Service-oriented systems:** understanding the 'register, find, bind' triangle, loose coupling, Software-as-a-Service, Governance issues Practical realization of service-oriented systems via web services, basics of xml and its use in web-service implementation, http protocol, utility of web-services ● **Basic web services stack:** understanding the SOAP protocol, WSDL, UDDI registry. Implementation of web services using the basic web services stack ● **Representational State Transfer (REST) web services:** implementation of RESTful web services, REST constraints, comparison of this approach of web-service implementation with that of the basic web-service stack, advantages and limitation of RESTful web services ● **Service composition:** understanding of the concepts of service orchestration and service choreography, static versus dynamic service composition, assessment of quality in service compositions, appropriate service selection for compositions, role of the customer in service composition
Suggested Books	Textbooks: 1. J. Snell, D. Tidwell, P. Kulchenko. ***Programming Web Services with SOAP***, O'Reilly, 2001. ISBN: 9780596000950 2. L. Richardson, S. Ruby, D. H. Hansson. ***RESTful Web Services***, O'Reilly, 2007. ISBN: 9780596529260 Reference books: 3. B. A. Christudas, M. Barai,V. Cacello, ***Service-Oriented Architecture with Java***, Packt Publishing, 2008. ISBN: 978-1847193216

Course Code	**CS 418/ CS 618**
Title of the Course	**Systems and Usable Security**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre requisite, if any	UG Level Courses on Operating Systems and Computer Networks
Objective(s)	● To understand the principles of systems security from an applied viewpoint and obtain hands-on experience on security threats and counter-measures. ● To study operating systems security, advanced topics on network security, access control and digital rights management, web security and usable security.
Course outcome	The student will have sound understanding of practical aspects of security and will be able to analyze and design the secure systems.
Course Syllabus	● **Introduction:** Computer Security Concepts, threats, Attacks, and Assets ● **Malicious Software:** Types of Malicious Software (Malware), Infected Content Viruses, Vulnerability Exploit Worms, Social Engineering SPAM E-mail, Trojans, System Corruption, Zombie, Bots, Information Theft Keyloggers, Phishing, Spyware, Stealthing Backdoors, Rootkits. ● **Operating System Security:** System Security Planning, Application Security, Linux/Unix Security, Windows Security, Virtualization Security ● **Access Control:** Access Control Principles, Subjects, Objects, and Access Rights, UNIX File Access Control, Role-Based Access Control, Attribute based Access Control. ● **Database Security:** The Need for Database Security, Database Management Systems, Database Access Control, Statistical Databases, Private Information Retrieval, Cloud Security. ● **Digital Rights Management:** Multicast security, copyright protection, Digital Fingerprinting. ● **Web Security:** Secure E-mail and S/MIME, Domain Keys Identified Mail, Secure Sockets Layer (SSL) and Transport Layer Security (TLS), HTTPS, IPv4 and IPv6 Security, Internet Authentication Applications, Kerberos, X.509, Public-Key Infrastructure, Federated Identity Management. ● **Wireless Security:** Wireless Security Overview, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security. ● **Usable Security:** Introduction to privacy, trust and semantic security, Visualizing privacy, Web browser security and privacy, Authentication and text passwords, biometrics and graphical passwords.

Suggested Books	Textbooks: 1. W. Stallings and L. Brown, ***Computer Security: Principles and Practice*** (2nd Edition), Prentice Hall, 2011. ISBN: 978-0132775069 2. A. Menezes, P. Oorschot, S. Vanstone, ***Handbook of Applied Cryptography***, Jaypee medical, 1996. ISBN: 978-0849385230 Reference books: 3. Goodrich and Tamassia, ***Introduction to Computer Security***, Addison-Wesley, 2011. ISBN: 978-0321512949. 4. Kaufman, Perlman, and Speciner, ***Network Security: Private Communication in a Public World***, (2nd edition), Prentice Hall, 2003. ISBN: 978-0130460196

Course Code	CS 422/ CS 622
Title of the Course	**Numerical Simulation**
Credit Structure	L-T- P-Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre requisite, if any	Calculus, Linear Algebra and Ordinary Differential Equations, Complex Analysis and Differential Equations, Numerical Methods
Objective(s)	• Simulation is a useful tool in almost all areas of engineering and science. • This course will introduce computational techniques for simulating applications from Electrical Engineering, Mechanical Engineering, Material Science, Physics, and Operations Research.
Course Syllabus	• **Problem Types:** Circuit analysis, structural analysis of automobiles, • analyzing drag force in aircrafts, engine thermal analysis, virtual environments for computer games, stock option pricing, electrostatic optimization for biomolecules etc. • **Equation Formulation Methods:** Stamping, node-branch, and nodal. • **Direct and Iterative Matrix Solution:** Error analysis, dense and sparse matrix factorizations, and Krylov methods. • **Nonlinear Systems Solution:** Multi-dimension Newton, fixed-point and functional iterations, and continuation schemes. • **Numerical Integration and Monte Carlo Methods:** Newton-Cotes, composite quadrature, Gauss quadrature, multiple integrals, generating samples, random tours, designing and analyzing random paths. • **Discretization Methods for Partial Differential Equations**: Finite difference, finite element, multi-grid, and spectral methods.
Suggested books	Textbooks: 1. G. Strang, ***Computational Science and Engineering***, Wellesley-Cambridge Press, 2007. ISBN: 978-0961408817 2. D. Kincaid and W. Cheney, ***Numerical Methods: Mathematics of Scientific Computing***, Brooks / Cole, 2007. ISBN: 978-0495114758 3. Y. Saad, ***Iterative Methods for Sparse Linear Systems***, SIAM, 2003. ISBN: 978-0898715347 4. C. T. Kelley, ***Solving Nonlinear Equations with Newton's Method***, SIAM, 2003. ISBN: 978-0898715460 Reference books: 5. E. L. Allgower and K. Georg, ***Introduction to Numerical Continuation Methods***, SIAM, 2003. ISBN: 978-0-89871-544-6

	6. G. S. Fishman, ***Monte Carlo Concepts, Algorithms, and Applications***, Springer, 1996. ISBN: 978-0387945279 7. W. L. Briggs, V. E. Henson, and S. F. McCormick, ***A Multigrid Tutorial***, SIAM, 2000. ISBN: 978-0-89871-950-5

Course Code	**CS 424**
Title of the Course	**Functional and Logic Programming**
Credit Structure	L-T- P-Credits 2-0-2-3
Name of the concerned Department	Computer Science and Engineering
Prerequisite, if any	Computer Programming
Objective(s)	• Functional-logic programming integrates most of the features of the classical declarative paradigms, namely of functional programming and of logic programming. • From the functional paradigm it inherits named or anonymous functions, nested expressions, efficient reduction strategies, higher-order functions, and types. • From the logic paradigm it inherits named relations, logical variables, partial data structures, unification, and built-in search.
Course Syllabus	• **Introduction to Functional and Logic Programming:** Overview of Declarative Programming, Basic Notions of Functional Programming, Basic Notions of Logic Programming, Basic Notions of Functional-Logic Programming. • **Terms in Functional and Logic Programming:** Taxonomy of Terms, Simple Terms, Complex Terms, Term Unification. • **Functional and Logic Definition Clauses:** Taxonomy and Syntax of Clauses, Logic Clauses, Functional Clauses, Functional-Logic Clauses. • **Higher Order Operations:** Function Composition, Compose as a higher order function, Relational Product as a higher order function. • **Case Study:** Study of a Functional-Logic Programming Language (e.g. Relational Functional Markup Language (RFML))
Suggested books	Textbooks: 1. J. Kelly, ***The Essence of Logic***, Prentice-Hall of India, 1997. ISBN: 978-0133963755 2. H.B. Enderton, ***Mathematical Introduction to Logic***, Academic Press, Elsevier, 2001. ISBN: 9780122384523 Reference books: 3. R. Wilensky, ***Common LISPcraft***, W. W. Norton and Co., 1986. ISBN: 978-0393955446 4. W. F. Clocksin and C.S. Melish, ***Programming in Prolo***g, Springer- Verlag, 2012. ISBN: 978-3540006787 5. G. Cousineau and M. Mauny, ***The Functional Approach to Programming***, Cambridge University Press, 1998. ISBN: 978-0521571838

Course code	**CS 425 / CS 625**
Title of the course	**Natural Language Processing**
Course Category	Departmental Elective / Institute Elective
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Familiarity with basic data structures and algorithms, ML concepts will help, though not necessary.
Objective(s)	The course is an introductory course in the natural language processing field. This is meant to get students familiar with the text processing techniques as well as more advanced techniques for text processing such as question answering, text summarization, parsing, semantic role labelling, etc.
Course Outcome	The students will learn: ● Basics of text processing and NLP ● Tokenization, lemmatization, word-embedding ● Applications of NLP to solve real-world problems.
Course Syllabus	● Introduction NLP tasks in syntax, semantics, and pragmatics. Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. A brief history of the field. ● N-gram Language Models The role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models. Part Of Speech Tagging and Sequence Labeling. Hidden Markov Models (Forward and Viterbi algorithms and EM training). ● Syntactic parsing Grammar formalisms and treebanks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs. ● Information Extraction (IE) Named entity recognition and relation extraction. IE using sequence labeling. word-sense disambiguation. Semantic Role Labeling. ● Machine Translation (MT) Basic issues in MT. Statistical translation, word alignment, phrase-based translation, and synchronous grammar. Advanced Concepts: Conversational Systems, chatbots.

Suggested Books	Textbooks: 1. D. Jurafsky and J. H. Martin, ***Speech and Language Processing***, Pearson Education, 2014. ISBN 9780133252934. 2. Manning and Schutze, ***Statistical Natural Language Processing***, MIT Press, 1999, ISBN 0262133601 Reference books: 3. J. Allen, ***Natural Language Understanding***, Pearson, 1994, ISBN: 978-0805303346 4. Y. Goldberg and G. Hirst, ***Neural Network Methods in Natural Language Processing***, Morgan and Claypool Publishers, 2017. ISBN: 978-1627052986

Course code	**CS 426 / CS 626**
Title of the course	**Foundations of Cyber-Physical Systems**
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematics, Automata Theory, Programming, Operating Systems
Objective(s)	• This course aims to introduce the basics of Cyber Physical Systems, distinguishing characteristics that sets them apart from their other embedded system incarnations. • The emphasis would be from systems specification, design and formal analysis perspectives.
Course outcome	The students will learn the basics of cyber-physical systems
Course Syllabus	• **Dynamical systems:** Continuous Vs Discrete behaviour, Hybrid behaviour, Reactive Systems, CPS key features, Synchronous Vs. Asynchronous paradigms. • **Control routine patterns:** Different control cycle actuations, Event/ Time -triggered, Static cyclic scheduling. • **Scheduling:** Realtime Scheduling policies, Rate-monotonic, Deadline- monotonic, Utilisation-based Schedulability. • **Kernels:** Real-time kernels and Kernel-based system development. • **Specification languages:** State machines (deterministic, nondeterministic), structural vs behavioural specification • **Correctness Analysis:** Requirements specification via Live, Safe and Precedence properties, Timed Processes and Protocols, Timed Automata, Hybrid Automata (Linear vs Non-linear), Mechanised Proof techniques, Deductive verification, 7. CPS Designs: From various application domains and CPS Case Studies.
Suggested Books	Textbooks: 1. A. Platzer, ***Logical Foundations of Cyber-Physical Systems***, Springer, Switzerland, 2017. ISBN 978-1-4419-8236-0 2. E. A. Lee and S. A. Seshia, ***Introduction to Embedded Systems: A Cyber-Physical Systems Approach***, Second Edition, MIT Press, Cambridge, 2017. ISBN 978- 0-262-53381-2 Reference books: 3. C. Baier and J. P. Katoen, ***Principles of Model Checking***, MIT Press, London, 2008. ISBN: 9780262026499

Course code	**CS 427/ CS 627**
Title of the course	**Advanced Computer Networks**
Credit Structure	L-T-P Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Computer Networks and Operating Systems
Scope of the Course	• This course will give a background on design principles of highperformance networking devices like switches and routers. • It will introduce advanced topics and recent trends in Computer Networks like Software Defined Networking, Data Center Networks, Information Centric Networking and Future Internet.
Course outcome	• Introduce PG/UG students to the recent advances and state-of-the art Computer Networking topics and develop understandings on the future research aspects in networking to help guide the students towards potential MTech/MS/PhD work. • Provide insights on the principles and design decisions behind networking aspects, Internet technologies and future research. • Expose students to hands-on exercises, work on network simulators and help them design and implement the networking protocols and applications.
Course Syllabus	• **High Performance Switching and Routing:** IP Address Lookup- Trie-based Algorithms, Hardware Lookup. Quality of Service – Need for Packet Classification, Different Classification Methods, TCAM based Classification, Differentiated Service, Traffic Polishing, Traffic Shaping, Packet Scheduling, Queue Management Techniques.Packet Switching: Switching Overview, Switching Fabric, Buffering in Switch Fabric, Multiplane and Multistage Switching Network • **Softwarization:** Software Defined Networking (SDN): Control and Data Plane Separation. Network Operating System (SDN Controllers), Intent based Networking (IBN), Southbound Interface (OpenFlow) and OpenVSwitch. Programmable Networking Devices, Network Virtualization: VirtIO, SR-IOV, and Network Function Virtualization (NFV). • **Data Centre Networking:** Networking Topologies: Fat-Tree, Clos, Leaf-Spine, Docker and Container Networking Interface (CNIs), Kubernetes. Switching in Data Centre Networking, Virtual Switches. High-Performance Computing Networks: HPC System Architectures, HPC Networking Standards, HPC Networking Software, Low Latency Ethernet (10 GbE, 40 GbE, 100 GbE, InfiniBand FDR). • **Named Data Networking:** Content Distribution on the Internet, Architectures for Information Centric Networking, Content Naming,

	Routing and Caching, Security in Named Data Networking.
Suggested Books	Textbooks: 1. H. J. Chao, B. Liu, ***High Performance Switches and Routers***, John Wiley and Sons, Inc, 2007. ISBN-10: 0-470-05367-4 2. G. M. de Brito, P. B. Velloso, and I. M. Moraes, ***Information-Centric Networks: A New Paradigm for the Internet***, Wiley-ISTE, 2013, ISBN: 9781848214491 Reference books: 3. Gary Lee, ***Cloud Networking: Understanding Cloud-based Data Centre Networks***, Morgan Kaufmann, 2014. ISBN: 139780128007280

Course code	CS 428/ CS 628
Title of the course	**Algorithmic Graph Theory**
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematical Structures, Data Structures and Algorithms.
Objective(s)	● This course shall impart basic background on the theoretical concepts of graph theory. ● The topics covered shall cover basic concepts and algorithmic aspects, such as graph representations, terminologies, and properties of different types of graphs, connectivity properties and algorithms, as well as some advanced topics, such as graph coloring, matching, planarity, and spectral graph theory.
Course outcome	The students will get firm foundations in solving real-world problems such as path optimization problems, and other graph-theoretical problems that are relevant to theoretical computer science and operations research.
Course Syllabus	● **Introduction to graphs, trees, and their properties:** Graphs, Representation of Graphs, Various Special Graphs, Walks, Graph Isomorphism, Spanning Trees, Counting Spanning trees in polynomial time, Algorithms for minimum weighted spanning trees. ● **Matching Algorithms and Cycles in Graphs:** Matching, Perfect matching, Augmenting path algorithm, Bipartite matching algorithm, Hall Marriage ● The Eulerian tour and Seven Bridges problem, Hamiltonian cycles and Travelling Salesman Problem, Necessary Conditions for Hamiltonian Graphs, Sufficient Conditions for Hamiltonian Graphs. ● **Coloring and Connectivity in graphs:** Vertex Coloring, Edge Coloring, Brook's theorem, Vizing Conjecture. Vertex and Edge Connectivity, Vertex- and edge-disjoint paths, testing connectivity, Algorithm for the cut-vertices, Algorithm for decomposing connected graph into blocks, Tutte's decomposition, edgec-onnectivity, Menger's Theorem. ● **Network Flow Algorithms:** Basic concepts on flows and networks, max- flow min-cut theorem, Ford-Fulkerson algorithm. ● **Planarity in graphs:** Planar graphs, Euler's Formula, Outer Planar Graphs, Kuratowski Theorem, Four Color Theorem. ● **Spectral graph theory:** Adjacency matrix, Laplacian matrix, Random regular graphs, Expander graphs,

	Ramanujan graphs. ● **Applications and Case studies:** Social Network Analysis, Complex ● Network Analysis.
Suggested Books	Textbooks: 1. D. B. West, ***Introduction to Graph Theory***, Pearson Education, 2015. ISBN: 8178088304. 2. R. Diestel, ***Graph Theory***, Springer-Verlag, 2000. ISBN: 0387950141. 3. R.B. Bapat, ***Graphs and matrices***, Springer, 2010. ISBN: 9789380250694. Reference books: 4. Bondy and U. S. R. Murthy, ***Graph Theory, Graduate Texts In Mathematics***, Springer, 2008. ISBN: 978-1-84628-969-9. 5. Alan Gibbons, ***Algorithmic Graph Theory***, Cambridge University Press, 1985. ISBN: 9780521288811. 6. T. Cormen and C.E. Leiserson and R.L. Rivest and C. Stein, ***Introduction to Algorithms***, The MIT Press, 2009. ISBN: 9780-262-03384-8 7. N. Deo, ***Graph Theory with Applications to Engineering and Computer Science***, PHI Learning, 1979. ISBN: 9788120301450

Course code	**CS 430/ CS 630**
Title of the course	**Data Center Networking**
Credit Structure	L-T-P- Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Students must have knowledge of UG-level computer network courses and have good programming skills in python and C/C++.
Scope of the Course	In this course, students are expected to learn the data center network architectures, their underlying protocols and understand the challenges faced in designing a data center.
Course outocme	The students will learn the data center network architectures, their underlying protocols.
Course Syllabus	● Introduction to network architectures Edge, Metro and Core Networks Evolution of Data Center Networks a switch-centric DCN, a modular DCN, a wireless DCN, and a hybrid DCN. ● Datacenter architectures and their building blocks Network fabric, cross connect design, Server addressing and routing protocols. Multipath routing. Overview of Software Defined Networking and Network Function Virtualization. ● State-of-the-art of data center networking architectures Fat tree, Helios, VL2, Portland, B-cube, Wavecube etc. ● Datacenter telemetry and resource management traffic engineering, congestion control, load balancing and resource allocation algorithms. Datacenter network performance enhancement strategies. ● Challenges involved in building a datacenter cross-connect and protocol scalability, fault tolerance, security, power etc. Recent advancements in data centers hybrid datacenter architectures etc..
Suggested Books	Textbooks: 1. Y. Liu, J. K. Muppala, M. Veeraraghavan, D. Lin, and M. Hamdi, ***Data Center Networks: Topologies, Architectures and FaultTolerance Characteristics***, Springer, 2013. ISBN: 9783319019482. Reference books: 2. Guo, D. ***Data center networking: Network topologies and traffic management in large-scale data centers***, Singapore, Springer, 2022. ISBN: 9789811693687. 3. M. Arregoces and M. Portolani, ***Data Center Fundamentals***, Cisco Press, 2004. ISBN: 1587050234. 4. L. Zhang and L. Chen, ***Cloud Data Center Network Architectures and Technologies***, CRC

	Press, 2021. ISBN: 9780367695705 5. J. Donovan and K. Prabhu, ***Building the Network of the Future: Getting Smarter, Faster, and More Flexible with a Software Centric Approach (100 Cases)***, CRC Press, 2017. ISBN: 9781138631526

Course Code	CS 432/ CS 632
Title of the Course	**Reinforcement Learning**
Credit Structure	L-T-P-Credit 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Linear algebra, probability and statistics, knowledge of algorithm design and analysis, and proficiency in programming.
Scope of the Course	• Learning the basic theory of reinforcement learning. • Understanding a range of reinforcement learning algorithms with their strengths and limitations. • Formulation of reinforcement learning problems for different applications.
Course Syllabus	● **Introduction:** Types of machine learning approaches: supervised, unsupervised, reinforcement; Elements of reinforcement learning: agent, environment, policy, reward and value functions ● **Multi-armed bandits:** Sequential decision making: exploration and exploitation; bandit problem solution: greedy, optimistic-greedy, and epsilon-greedy algorithms; Upper confidence bound bandit algorithm. Markov Processes: Markov property, Markov chains, Markov reward process, Markov decision process (MDP), Bellman expectation equation, optimal value function, optimal policy, Bellman optimality equation. Dynamic Programming: Dynamic programming for MDP, iterative policy evaluation, policy improvement, policy and value iteration. ● **Monte Carlo Methods:** Model free reinforcement learning, Monte Carlo policy evaluation and estimation of action values, on- and off-policy ● **Temporal-Difference Learning:** Temporal-Difference (TD) prediction, TD(0), TD(1), TD(λ), TD control methods: SARSA, Q-Learning and variants. ● **Function Approximation Methods:** Risk minimization, eligibility trace for function approximation, value function approximation (VFA), Monte Carlo learning and TD learning for policy evaluation with linear VFA. ● **Policy Gradients:** Policy gradient methods, policy search methods, gradient-free methods, finite difference methods, likelihood ratio policy gradient, bias and variance in reinforcement learning, actor-critic methods.
Suggested Books:	Textbooks: 1. R. S. Sutton and A. G. Barto, ***Reinforcement Learning - An Introduction***, MIT Press, Cambridge, 1998. ISBN: 978-0262193986 Reference Books:

	2. P. Winder, ***Reinforcement Learning: Industrial Applications of Intelligent Agents***, O'Reilly Media, Inc, 2020. ISBN: 978-1098114831 3. K. P. Murphy, ***Machine Learning: A Probabilistic Perspective***, The MIT Press, 2012. ISBN: 978-0262018029 4. I. Gridin, ***Practical Deep Reinforcement Learning with Python***, BPB Publication, 2022. ISBN: 978-9355512055

Course code	**CS 435/ CS 635**
Title of the course	**Deep Learning**
Credit Structure	L - T - P - Credits 2-0-2-3
Name of the Concerned Discipline	Computer Science and Engineering
Pre-requisite, if any	Linear algebra, probability, optimization
Objective(s)	● This is an introductory course in the field of deep learning. ● The course will cover latest advancement in the field of deep learning.
Course outcome	The students will learn the basics of deep learning and its applications.
Course Syllabus	● History of Deep Learning, Deep Learning Success Stories ● McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm. Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent, Feedforward Neural Networks. FeedForward Neural Networks, Backpropagation. Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp. ● Autoencoders and relation to PCA, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders. Regularization: Bias Variance Tradeo , L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying. ● Greedy Layerwise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization. Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet. Learning Vectorial Representations of Words. Recurrent Neural Networks, Backpropagation through time. Encoder Decoder Models, Attention Mechanism.
Suggested Books	Textbooks: 1. Ian Goodfellow,Yoshua Bengio, and Aaron Courville, ***Deep Learning***, MIT Press, 2016. ISBN: 978-0262035613 2. Li Deng and Dong Yu, ***Deep Learning Methods and Applications***, NOW Publishers, 2014. ISBN : 978-1601988140 Reference books: 3. Charu C. Aggarwal, ***Neural Networks and Deep Learning: A Textbook***, Springer, 2018. ISBN: 978-3319944623 4. Eugene Charniak, ***Introduction to Deep Learning***, MIT Press, 2019. ISBN: 978-0262039512

Course Code	**CS 438/ CS 638**
Title of the Course	**Network Softwarization and Management**
Course Category	Departmental Elective
Credit Structure	L-T-P–Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite if any	Knowledge of Computer Networks
Course Objective	This course will take a retrospective look at how network softwarization evolved in the past decade, discuss its current promises, enablers, and challenges, and finally speculate what direction it will take in the future.
Course Outcomes	Students will get an idea of how modern telecom networks are being softwarized and also understand the key design principles and obstacles in the process through hands-on experience.
Course Content	● Turmoil in the telecom industry: need for network softwarization, Service provider network design, Architecture of forwarding elements, Ethernet evolution: Carrier Ethernet, IP/MPLS networks and traffic engineering, Segment routing. ● Software-defined networking (SDN), SD-WANs, SDN architectures in optical and data-center networks, Data-plane programmability, Network Virtualization – VXLAN and NVGRE, SDN experiments. ● Network Function Virtualization (NFV): vision and obstacles, Edge/Cloud-native networking, Application of virtualization in designing next-generation cellular networks (5G and beyond) – vRANs, Network slicing. ● Modern network management paradigms, OpenConfig, AI-assisted network management, Self-driving and Zero-touch networking, Network digital twins, Measurements in softwarized networks. ● Impact of softwarization on core, edge, and access networks, Verification/debugging of softwarized networks, Availability/Reliability and security challenges, Rise of the open-source ecosystem, Softwarization status of a few service provider networks.
Suggested Books	Textbooks: 1. J. Donovan and K. Prabhu, ***Building the Network of the Future, Getting Smarter, Faster, and More Flexible with a Software Centric Approach***, Chapman and Hall/CRC, 2017, ISBN: 978-1138631526 2. W. Stallings, ***Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud***, Pearson Education, 2016, ISBN: 978-9332573864 Reference books:

	3. G. Varghese and J. Xu, ***Network Algorithmics An Interdisciplinary Approach to Designing Fast Networked Devices***, Morgan Kaufmann, 2022. ISBN: 978-0128099278

Course code	**CS 440/640**
Title of the course	**Distributed Network Algorithms**
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Discrete Mathematics, Data Structures and Algorithms, Design and Analysis of Algorithms
Course objective	● This course will cover the fundamentals of distributed network algorithms. ● With the rise and evolving characteristics of Internet and blockchain systems, along with paradigm shift from single processor computing to multi-processor computing, it is imperative to understand the fundamentals of distributed network algorithms. ● The students will gain knowledge of computation mechanisms of distributed systems, and how they are different from the computation theory of single processor computation.
Course outcome	● Students will be able to model a distributed computing problem and design algorithm to address the problem. They will also be able to prove the correctness of the algorithm and analyse its complexity. ● They will also be able to design and analyse distributed algorithms for real-life distributed systems.
Course Syllabus	● **Models:** Distributed network models, Performance of distributed algorithms: complexity measures: time complexity, message complexity. ● **Distributed Broadcast, Shortest Path and Tree Algorithms:** Broadcast, Lower bound for broadcast, Tree Broadcast, Convergecast on tree, Upcast, Downcast, Constructing a BFS Tree, Information spreading, A global distributed shortest path algorithm, Distributed Bellman Ford algorithm. ● **Leader Election and Distributed Consensus:** Leader election in ring network, Leader election in a complete network, Leader election in general distributed network, Consensus in synchronous and asynchronous distributed system, Consensus under fault, Byzantine generals problem, Paxos algorithm. ● **Local Symmetry Breaking:** Maximum independent set, Coloring, Deterministic symmetry breaking, ● **Minimum Spanning Tree:** Gallagar-Humblet-Spira (GHS) algorithm, Pipeline Algorithm, Garay-Kutten-Peleg (GKP) Algorithm

Suggested Boo	Textbooks: 1. D. Peleg, ***Distributed Computing A Locality Sensitive Approach***, Society for Industrial and Applied Mathematics, 2000. ISBN: 978-0898714647 2. N. Lynch, ***Distributed Algorithms***, Morgan Kaufmann Publishers, 1996, ISBN: 978-1558603486 Reference books: 3. M. Singhal and N. G. Shivaratri, ***Advanced Concepts in Operating Systems***, McGraw Hill Education, 2017, ISBN: 978-0070472686

Course code	**CS 442/642**
Title of the course	**Algebraic Graph Theory**
Course Category	Departmental Elective
Credit Structure	L - T - P – Credits 2-0-2-3
Name of the Concerned Department	Computer Science and Engineering, IIT Indore
Pre-requisite, if any	The basic knowledge of Linear algebra and Graph theory
Scope of the course	This course is for research directions in the area of spectral graph algorithms for example complex network analysis, theoretical computer science.
Outcome of the course	Students will be able to perform the analysis of graph properties using matrix theory. They will gain the skill to formulate and derive the bounds on some intractable graph problems.

Course Syllabus	**Module 1**: Matrices associated with graphs, Adjacency matrix, Laplacian matrix, Distance matrix, Seidel Matrix, Spectral Theorem **Module 2**: Finding number walks, connected components, Counting number of spanning trees, Matrix-Tree Theorem(s) **Module 3**: Algebraic connectivity, Regular graphs, Random walks, expanders, Ramanujan Graphs **Module 4**: Graph Isomorphism problem, Graphs determined by the eigenvalues. **Module 5**: Strongly regular graphs, Friendship Theorem, Spectral bounds on NP-hard problems on graphs
Suggested Books	**Textbook:** 1. C. Godsil, G. Royle : *Algebraic Graph Theory* : (Springer) : London : 2001: 0387952411 2. R. B. Bapat : *Graphs and Matrices* : Hindustan Book Agency : New Delhi : 2014 : 9789380250694 **References:** 1. A. E. Brouwer, W. H. Haemers : *Spectra of graphs* : (Springer) : London : 2011 : 9781461419396 2. F.R.K. Chung : *Spectral graph theory* : American Mathematical Society: UK: 1997: 978-0821803158

Course Code	**CS 444/ CS 644**
Title of the Course	**Advanced Blockchain**
Course Category	Departmental Elective
Credit Structure	L-T-P–Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Prerequisite if any	Discrete Mathematics, Data Structure and Algorithms
Course Objective	This course will offer in-depth understanding of theoretical underpinnings, applications, best practices and research activities in the domain of Blockchain.
Course Outcomes	Students will develop a clear understanding of state-of-the-art Blockchain Technology, open research challenges and future directions.
Course Content	• Introduction to Blockchain Stacks and its layers, synchronous model and the Dolev-Strong protocol for Byzantine broadcast, asynchronous, partially synchronous model, PBFT, Tendermint and Hotstuff protocol • Deep Dive into Bitcoin, Longest Chain consensus and Formal security guarantees of Proof-of-Work, Selfish Mining Attack and Incentive Compatibility • Longest Chain vs Byzantine Fault Tolerance, Ouroboros, Ouroboros-Praos, Algorand, Availability - Finality Dilemma & Finality Gadgets • Security and Privacy in Blockchain, Introduction to Monero and ZCash • Scalability in Blockchain: Lightning Network, Sidechains, Rollups, Sharding
Suggested Books	Textbooks: 1. R. Wattenhofer, ***Blockchain Science: Distributed Ledger Technology***. Inverted Forest Publishing, 2019, ISBN: 9781793471734 2. A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder. ***Bitcoin and cryptocurrency technologies: a comprehensive introduction***. Princeton University Press, 2016. ISBN: 9780691171692 3. V. Gramoili, ***Blockchain Scalability and its Foundations in Distributed Systems***, Springer, 2022. ISBN 978-3-031-12577-5 Reference books:

	4. Serhack, ***Mastering Monero: The future of Private transactions***, 2018, ISBN: 978-1731079961 5. A. M. Antonopoulos, O. Osuntokun, and R. Pickhardt, ***Mastering the lightning network***, O'Reilly Media, Inc., 2021, ISBN: 9781492054863

Course code	**CS 446/ CS 646**
Title of the course	**Blockchain for Responsible Computing**
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the concerned Department	Computer Science & Engineering
Pre-requisite, if any	Programming and Basic knowledge of Software Engineering
Course objective	● To understand trust and accountability stack with blockchain technology. ● To understand the ethical and societal implications of blockchain's features. ● To explore and understand the responsible computing practices with blockchain.
Course outcome	Students gain implementing skills with applied blockchain tech for the needed responsibility computing. Students will explore the blockchain fabric as distributed technology to enhance trust, transparency and accountability across domains.
Course Syllabus	● Centralized vs Decentralized Computing, Local vs Distributed systems, On premises vs cloud instances, Foundations of Blockchain, Decentralized ledger, Consensus ● Privacy and data protection Considerations in Blockchain, Challenges in Blockchain, Smart contracts, Protection against fraud and cyber-vulnerabilities ● Blockchain and Sustainable Development Goals, Blockchain in Supply Chain/I4.0/Health Applications -Trust brokerage, Traceability, transparency, and responsible sourcing_ Ethical/privacy and fairness considerations ● Blockchain development, Creating and deploying smart contracts, Debugging and security testing, Responsible coding practices in blockchain ● Regulatory and Legal Considerations with Blockchain, Compliance with legal requirements of the land, Case Studies and Industry Trends, Real projects guidelines
Suggested Books	Textbooks: 1. SK H. Islam, A. Kumar Pal, D. Samanta, S. Bhattacharyya, ***Blockchain Technology for Emerging Applications***, Elsevier, 2022. ISBN: 9780323901949 2. A. Lipton and A. Treccani, ***Blockchain And Distributed Ledgers : Mathematics, Technology, And Economics***, World Scientific Publishing Co Pte Ltd, 2021. ISBN: 978-9811221514

	Reference books: 3. I. Bashir, ***Mastering Blockchain***, Packt Publishing, 2020. ISBN: 978-1839213199

Course Code	CS 448/648
Title of the Course	Software Verification and Certification
Course Category	Department Elective
Credit Structure	L-T-P–Credits 2-1-0-3
Department	Computer Science and Engineering
Pre-requisite	Knowledge of Automata Theory and Logic, Computer Programming
Objectives	1. To understand the significance of correctness, safety, and security in open-source software. 2. To learn techniques of formal verification and static code analysis.
Outcomes	1. Apply formal verification methods to ensure software safety, liveness, and correctness. 2. Embed correctness checks throughout the software lifecycle for certified and responsible development.
Course Syllabus	• **Foundations of Software Correctness:** Fundamentals of software correctness-safety-liveness, formal software verification, verification techniques: deductive vs algorithmic and hybrid approaches • **Formal Verification Techniques:** Model checking and Theorem proving, Program analysis: Static vs dynamic, Case studies, synergistic verification and validation • **Verification- Languages, Tools, and Standards:** property specification languages, system modelling languages, and tools, domain specific Coding standards and implications • **Collaborative Verification and Licensing Considerations:** openness vs verifiability, Industrial case studies, Application in responsibility specification and analysis • **Correctness-Driven Software Lifecycle:** Integration of correctness in the software development lifecycle, correctness-certification of open-source projects
Suggested Books	Textbooks: 1. Edmund M. Clarke Jr., Orna Grumberg, Daniel Kroening, Doron Peled and Helmut Veith, "**Model Checking",** the MIT Press, 2018, ISBN: 9780262038836 2. Xavier Rival, Kwangkeun Yi, "**Introduction to Static Analysis: An Abstract Interpretation Perspective",** The MIT Press, 2020, ISBN:9780262043410 Reference books: 3. Steven Weber, "**The Success of Open Source"**, Harvard University Press, USA, 2005, ISBN: 9780674018587

Course code	**CS 450/CS 650**
Title of the course	**Generative AI**
Course Category	Department Elective
Credit Structure	L - T - P - Credits 2-1-0-3
Name of the Concerned Department	Computer Science and Engineering
Pre-requisite, if any	Machine Learning
Objective(s)	● This course is designed to provide an overview of Generative AI. ● Gain foundational knowledge of LLMs, practical skills, and a functional understanding of how generative AI works.
Course Outcomes	● Students will dive into the latest research on Generative AI to understand how companies are creating value with cutting-edge technology. ● They will learn the best practices of prompt engineering and fine-tuning LLMs.
Course Content	● Machine Learning Basics: Classification and regression, supervised, unsupervised, and self-supervised. ● Introduction to the large language model (LLMs), large vision models (LVM), large speech model (LSM). ● Prompt Engineering: In-context learning, types of prompting— chain of thoughts, tree-of thoughts, Retrieval-augmented generation. ● LLM Fine-Tuning—Need for fine-tuning, instruction fine-tuning and parameter efficient fine-tuning ● Building App with LLMs, Benchmarking the LLMs, Security and Legal implications.
Suggested Books	Textbooks: 1. J. Alammar and M. Grootendorst, ***Hands-On Large Language Models***, O'Reilly Media Inc., 2024. ISBN: 9781098150969 (In Print) Reference books: 2. S. Kublik, and S. Saboo, ***GPT-3: Building Innovative NLP Products Using Large Language Models***, O'Reilly Media, 2023. ISBN: 978-1098113629

Course Code	**CS 454/ CS 654**
Title of the Course	**Distributed Algorithms for Robots and Agents**
Credit Structure	L - T - P – Credits 2 - 1 - 0 - 3
Department	Computer Science and Engineering
Pre-requisites	Basic Knowledge of Discrete Mathematics, Design and Analysis of Algorithms
Course Objective	• Introduce algorithmic foundations of distributed computing by mobile agents and swarm robots. • Analyze the impact of synchrony, anonymity, visibility, and faults on solvability. • Analyze pattern formation problems theoretically and with simulation.
Course Outcome	• Model distributed problems in geometric and graph-based environments. • Prove correctness and impossibility results of distributed algorithms. • Design, simulate, and analyze coordination algorithms for multi-robot systems.
Course Syllabus	**Models and Fundamentals** - Look-Compute-Move cycles; synchronous vs asynchronous models. - Memory and visibility: oblivious vs luminous robots. - Mobile agents on graphs; complexity and performance measures. **Coordination Algorithms** - Rendezvous, gathering, and leader election. - Symmetry breaking; impossibility results. - Communication and coordination under minimal assumptions. **Pattern Formation and Fault Tolerance** - Arbitrary and symmetric pattern formation; uniform circle problem. - Effects of limited visibility and robot lights. - Fault-tolerance: crash and Byzantine models. **Graph and Advanced Algorithms** - Gathering and exploration of graphs; use of tokens or pebbles. - Search, evacuation, and patrolling in dynamic networks. - Decontamination and programmable matter.

Suggested Books	**Textbooks:** 1. Nicola Santoro and Paola Flocchini, Distributed Computing by Oblivious Mobile Robots, Morgan & Claypool, 2012. ISBN: 978-1608456864. 2. David Peleg, Distributed Algorithms: A Locality Sensitive Approach, SIAM, 2000. ISBN: 978-0898714647. **Reference Books:** 3. Nancy A. Lynch, Distributed Algorithms, Morgan Kaufmann, 1996. ISBN: 978-1558603486. 4. Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, Distributed Computing by Mobile Entities: Current Research in Moving and Computing, Springer, 2019. ISBN: 978-3-030-11071-3.

Course Code	**CS 456/ CS 656**
Title of the Course	**Software-defined IoT**
Course Category	Departmental Elective
Credit Structure	L - T - P – Credits 2 - 0 - 2 - 3
Department	Computer Science and Engineering
Course Instructor	
Pre-requisites	Knowledge of the basics of computer networks
Course Objective	• Introduce advanced components of IoT networking and communication. • Explore the role of software-defined networking (SDN) in IoT. • Understand data handling, analytics, and data management in IoT systems.
Course Outcome	• Understand advanced concepts and functionalities of IoT devices and networking. • Design and manage IoT-based prototypes focusing on data handling and analytics.
Course Syllabus	**Introduction to IoT and IoT Networking** • IoT architecture, components, and ecosystem. • Sensor networks, edge and fog computing. • IoT networking challenges: scalability, latency, and security. **IoT Communication Protocols and Their Applications** • Overview of IoT communication stack. • MQTT, CoAP, LoRaWAN, and NB-IoT. • Protocol selection for different IoT domains. **Software-Defined Networking (SDN) for IoT** • SDN fundamentals and layered architecture. • OpenFlow protocol, Northbound and Southbound APIs. • SDN controllers and IoT use cases. **Data Handling, Analytics, and Management in Software-defined IoT (S-IoT)** • Data collection and edge analytics. • Cloud and edge data management. • Security, privacy, and interoperability in IoT data. **Typical applications of S-IoT** • Smart cities and industrial automation. • Precision agriculture and healthcare systems. • Edge AI and digital twin applications. **List of Representative Experiments** • Developing Secure MQTT Communication for IoT Devices. • Implementing a LoRaWAN Sensor Network and Gateway

	System. • Developing Edge Analytics for Real-Time Anomaly Detection. • Implementing SDN-Based Traffic Prioritization in IoT Networks. Developing Secure Data Management and Access Control in S-IoT.
Suggested books	**Textbooks** 1. P. Goransson, C. Black, and T. Culver, *Software Defined Networks: A Comprehensive Approach*, Morgan Kaufmann, 2016. ISBN: 978-0128045558. 2. S. Misra, A. Mukherjee, and A. Roy, *Introduction to IoT*, Cambridge University Press, 2022. ISBN: 978-1108959742. 3. S. Misra, C. Roy, and A. Mukherjee, *Introduction to Industrial Internet of Things and Industry 4.0*, Cambridge University Press, 2020. ISBN: 978-1032146751. **Reference Books** 4. W. Stallings, *Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud*, Pearson Education, 2016. ISBN: 978-0-13-417539-3.