CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

FIFTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2018 - 2019 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING

(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41

Fax: 0452 2483427

Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E Degree (Computer Science and Engineering) Programme

COURSES OF STUDY
(For the candidates admitted from 2018- 19 onwards)

FIFTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			Credits
			L	T	P	
THEORY						
18CS510	Numerical Methods	ES	2	1	-	3
18CS520	Theory of Computation	PC	3	-	-	3
18CS530	Data Communication and Networks	PC	3	-	-	3
18CSPX0	Program Elective – I	PE	3	-	-	3
18CSGX0	General Elective – I	GE	3	-	-	3
THEORY CUM PRACTICAL						
18CS560	Software Engineering: Theory and Practice	PC	2	-	2	3
PRACTICAL						
18CS570	Databases Lab	PC	-	-	2	1
18CS580	Network Programming Lab	PC	-	-	2	1
18ES590	System Thinking	ES	1	-	2	2
18CHAC0	Essence of Indian Knowledge	AC	2	-	-	-
Total			**19**	**1**	**8**	**22**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
PC : Program Core
PE : Program Elective
OE : Open Elective

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E Degree (Computer Science and Engineering) Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2018-19 onwards)

FIFTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment	Terminal Exam *	Max. Marks	Terminal Exam	Total
THEORY								
1	18CS510	Numerical Methods	3	50	50	100	25	50
2	18CS520	Theory of Computation	3	50	50	100	25	50
3	18CS530	Data Communication and Networks	3	50	50	100	25	50
4	18CSPX0	Program Elective – I	3	50	50	100	25	50
5	18CSGX0	General Elective – I	3	50	50	100	25	50
THEORY CUM PRACTICAL								
6	18CS560	Software Engineering: Theory and Practice	3	50	50	100	25	50
PRACTICAL								
7	18CS570	Databases Lab	3	50	50	100	25	50
8	18CS580	Network Programming Lab	3	50	50	100	25	50
9	18ES590	System Thinking	3	50	50	100	25	50
10	18CHAC0	Essence of Indian Knowledge	-	50	50	100	25	50

* Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks

		Category	L	T	P	Credit
18CS510	**NUMERICAL METHODS**	ES	2	1	0	3

Preamble

Numerical method deals with finding approximate solutions of polynomial, simultaneous algebraic equations, Interpolation, Differentiation and Integration, ODEs and PDEs by various Numerical techniques. The course is designed to impart the knowledge and understanding of the above concepts to computer science engineering students and apply them in their areas of specializations.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Apply appropriate numerical methods to solve algebraic, transcendental equations and polynomial equations.	10
CO2	Compute an approximate solution for simultaneous linear algebraic equations and the inverse of a non-singular matrix.	15
CO3	Calculate the approximate solutions for problems related to interpolation, differentiation and integration	25
CO4	Apply various predictor corrector methods for finding approximate solutions of Ordinary Differential Equations	20
CO5	Apply finite difference method to solve Boundary value problems.	10
CO6	Classify PDE and Apply various computational methods for finding approximate solutions of Partial Differential Equations of different types	20

*** Weightage depends on Bloom's Level, number of contact hours

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value		1.1.1, 2.1.1
CO2	TPS3	Apply	Value		1.1.1, 2.1.1
CO3	TPS3	Apply	Value		1.1.1, 2.1.1
CO4	TPS3	Apply	Value		1.1.1, 2.1.1
CO5	TPS3	Apply	Value		1.1.1, 2.1.1
CO6	TPS3	Apply	Value		1.1.1, 2.1.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		L	L		L	L	L		L	M		L
CO2	S	M	L		L	L		L	L	L		L	M		L

CO3	S	M	L		L	L		L	L	L		L	M		L
CO4	S	M	L		L	L		L	L	L		L	M		L
CO5	S	M	L		L	L		L	L	L		L	M		L
CO6	S	M	L		L	L		L	L	L		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continious Assessment Tests			Assignment			Terminal Examinations
	1	2	3	1	2	3	
Remember	10	10	10	-	-	-	-
Understand	30	30	30	-	-	-	30
Apply	60	60	60	100	100	100	70
Analyse							
Evaluate							
Create							

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Assignment/Practical Component
Perception	
Set	
Guided Response	
Mechanism	
Complex Overt Responses	
Adaptation	
Origination	

Sample Questions for Course Outcome Assessment**

** (2 to 3 at the cognitive level of course outcome)

Course Outcome 1 (CO1)

1 Predict the positive root of $3x-\sqrt{1+Sin\,x}=0.$ by fixed point method.

2 Compute the negative roots of $2x^3-7x^2+7x-2=0.$ by Bisection method

3 Compute the positive root of $x-Cos\,x=0.$ using Newton's Raphson method.

Course Outcome 2 (CO2)

1 Calculate the approximate solution to the system of equations by Gauss Seidel method.
$x+3y+10z=24\,;2x+17y+4z=35\,;28x+4y-z=32.$

2 Calculate the approximate solution to the system of equations by Gauss Jacobi method.
$8x+y+z=8\,;2x+4y+z=4\,;x+3y+3z=5.$

3 Calculate the inverse of the matrix $\begin{pmatrix}1 & 6 & 1\\ 1 & 2 & 0\\ 0 & -0 & 3\end{pmatrix}$ by Gauss Elimination method.

Course Outcome 3 (CO3)

1 Evaluate the integral $\int_1^2 \frac{dx}{1+x^2}$ using Trapezoidal rule with two subintervals

2 Use Runge Kutta method to compute y for x =0.1, given $y^1 = \frac{xy}{1+x^2}$, y(0)=1, take h = 0.1

3 For the following data, calculate the differences and obtain the forward and backward differene polynomials. Interpolate at x=0.25 and x=0.35

x	0.1	0.2	0.3	0.4	0.5
f(x)	1.40	1.56	1.76	2.00	2.28

Course Outcome 4 (CO4)

1 Calculate $y(0.8)$ using Milne's predictor corrector method given that $y' = y - x^2$; $y(0) = 1;\ y(0.2) = 1.12; y(0.3) = 1.46;\ y(0.6) = 1.73$

2 Calculate $y(0.8)$ using Adam's predictor corrector method given that $y' = xy + y^2$; $y(0) = 1;\ y(0.2) = 1.12; y(0.3) = 1.46;\ y(0.6) = 1.73$

3 Using Milne's method, find y(2) if y(x) is the solution of $y^1 = \frac{1}{2}(x+y)$, given y(0)=2, y(0.5)= 2.636, y(1)= 3.595 and y(1.5)= 4.968

Course Outcome 5 (CO5)

1 Solve the boundary value problem $x^2 y'' = 2y - x,\ 2 < x < 3, y(2) = 0, y(3) = 0$ using the Numerov method with h=1/3

2 Solve y″ + y = 2 with condition y (0) = 0, y (1) = 1, using Hermitian Method

3 Using Finite difference method Solve y″ - 64y + 10 = 0, $x \in (0,1)$,y (0) = y(1) = 0 by sub dividing the internals into i) Four equal parts ii) Two equal parts

Course Outcome 6 (CO6)

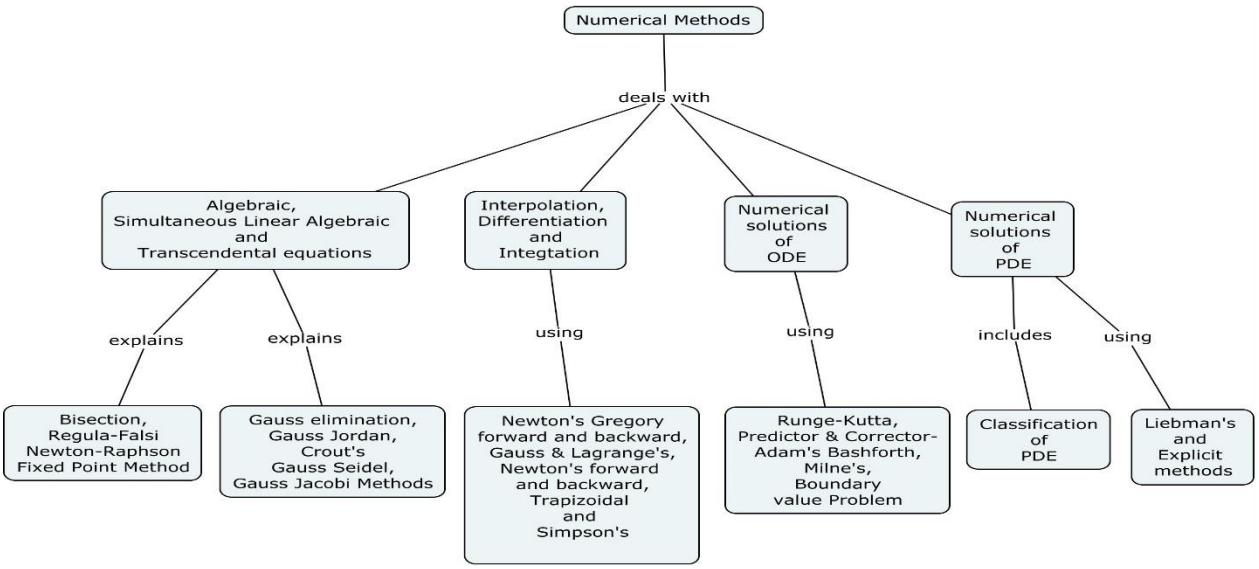
1 Solve: $u_{xx} + u_{yy} = 0$; over the square mesh of side 4 satisfying the following boundary conditions:

$u(0, y) = 0$; $0 \le y \le 4$; $u(4, y) = 12 + y$; $0 \le y \le 4$; $u(x, 0) = 3x$; $0 \le x \le 4$; $u(x, 4) = x^2$; $0 \le x \le 4$.

2 Using Bender Schmidt method find the solution of the parabolic equation $\frac{\partial^2 u}{\partial x^2} - 2\frac{\partial u}{\partial t} = 0$; *where* $u(0,t) = 0 = u(4,t)$; $u(x,0) = x(4-x)$. Assume h=1.Calculate the values up to t=5.

3 Solve numerically, $4u_{xx} = u_{tt}$ with boundary conditions u (0,t)=0, u(4,t)=0 and the initial conditions $u_t(x,0)=0$ and u(x,0)= x(4-x), taking h =1. (for 4 time steps)

Concept Map



Syllabus

Algebraic, Simultaneous linear Algebraic and Transcendental Equations: Bisection, Regula falsi, Newton-Raphson, Fixed point Method, Gauss Elimination, Gauss Jordan method, Crout's method ,Gauss Seidel and Gauss Jacobi methods, Inversion by Gauss Jordan method.

Interpolation, Differentiation and Integration: Newton Gregory's forward and backward difference interpolation formulae, Gauss's and Lagrange's interpolation formulae, Newton's forward and backward formulae for derivatives, Trapezoidal, Simpson's 1/3rd and 3/8th rules.

Numerical solutions of ODE: Runge Kutta Method of fourth order, Predictor-Corrector Method- Adams Bashforth, Milne's Method, Boundary Value Problem- Solution by Finite difference method.

Numerical solutions of PDE: Classification of PDE, Solution of Elliptic equations by Leibmann's method, Solution of parabolic and Hyperbolic equations by explicit methods.

Learning Resources

1. Sastry S.S "Introductory Methods of Numerical Analysis" Fifth edition Prentice Hall of India , New Delhi -2006
2. Iyengar.S.R.K, Jain.R.K., "Numerical Methods for Scientific and Engineering Computation"-Fifth edition, New Age International Publishers, New Delhi-2009
3. B.S.Grewal," Numerical Methods",- Nineth Edition- Khanna Publishing Company-New Delhi -2010.
4. Steven C. Capra, "Applied Numerical Methods with Matlab for Engineers and Scientists" –third edition, The McGraw-Hill Companies- 2012.
5. Steven C. Capra, Raymond P.Canale, "Numerical Methods for Engineers"- fifth edition, The McGraw-Hill Companies- 2006.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcomes
1	**Algebraic, simultaneous linear algebraic and Transcendental equations**		
1.1	Bisection , Regula-falsi method	2	CO1
1.2	Newton- Raphson method	1	CO1
	Tutorial	1	CO1
1.3	Fixed point method	1	CO1
1.4	Gauss elimination and Gauss Jordan methods	2	CO2
	Tutorial	1	
1.5	Crout's method, Gauss Jacobi and Gauss Siedal methods	2	CO2
1.6	Inversion by Gauss Jordan method	1	CO2
	Tutorial	1	
2	**Interpolation, Differentiation and Integration**		
2.1	Newton Gregory's forward and backward difference interpolation formulae	2	CO3
2.2	Gauss's and Lagrange's interpolation formulae	1	CO3
	Tutorial	1	
2.3	Newton's forward and backward formulae for derivatives	1	CO3
2.4	Trapezoidal, Simpson's 1/3rd and 3/8th rules	1	CO3
	Tutorial	2	
3	**Numerical Solution of ODE**		
3.1	Runge Kutta Method of fourth order	1	CO4
3.2	Predictor-Corrector Method- Adams Bash forth, Milne's Method	2	CO4
	Tutorial	2	
3.3	Boundary value problem – Solution by finite difference method	2	CO5
	Tutorial	1	
4	**Numerical Solution of PDE**		
4.1	Classification of PDE	1	CO6
4.2	solution of elliptic equations by Leibmann's Method	2	CO6
	Tutorial	1	
4.3	Solution of parabolic and Hyperbolic equations by explicit methods	2	CO6
	Tutorial	2	
	TOTAL	36	

Course Designers:

1. Dr.A.Anitha anithavalli@tce.edu
2. Ms. H.Sri Vinodhini srivinodhini@tce.edu

18CS520	THEORY OF COMPUTATION	Category	L	T	P	Credit
		PC	3	0	0	3

Preamble

This course will introduce students to three foundational areas of computer science namely the basic mathematical models of computation, problems that can be solved by computers and problems that are computationally hard. It also introduces basic computation models, their properties and the necessary mathematical techniques to prove more advanced attributes of these models. The students will be able to express computer science problems as mathematical statements and formulate proofs.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Construct the abstract machines including finite automata, pushdown automata, and Turing machines from their associated languages	15
CO2	Make use of pumping lemma to show that a language is not regular / not context-free	15
CO3	Construct finite automata, pushdown automata, Turing machines for the given grammar	20
CO4	Construct the grammar for any given finite automata, pushdown automata or Turing machines	20
CO5	Outline the characteristics of P, NP and NP Complete problems in the context of Turing machines	15
CO6	Illustrate the unconventional model of computation associated with a new computing paradigms including DNA and Membrane Computing	15

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2,2.3.2,3.2.3,4.4.3
CO2	TPS3	Apply	Value	Mechanism	1.2,2.3.2, 3.2.3,4.4.3

CO3	TPS3	Apply	Value	Mechanism	1.2,2.3.2, 3.2.3,4.4.3
CO4	TPS3	Apply	Value	Mechanism	1.2,2.3.2, 3.2.3,4.4.3
CO5	TPS2	Understand	Respond	Guided Response	1.2,2.3.2, 3.2.3,4.4.3
CO6	TPS2	Understand	Respond	Guided Response	1.2,2.3.2, 3.2.3,4.4.3

Mapping with Programme Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L					M	L	M		L	M		L
CO2	S	M	L					M	L	M		L	M		L
CO3	S	M	L					M	L	M		L	M		L
CO4	S	M	L					M	L	M		L	M		L
CO5	M	L						L		L			L		
CO6	M	L			L	L		L		L			L	L	

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	2	3	1	2	3	
Remember	20	20	20	-	-	-	20
Understand	30	40	30	30	30	30	30
Apply	50	40	50	70	70	70	50
Analyse	0	0	0	-	-	-	-
Evaluate	0	0	0	-	-	-	-
Create	0	0	0	-	-	-	-

Course Outcomes Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Assignment/Practical Component
Perception	-
Set	-

Guided Response	30
Mechanism	70
Complex Overt Responses	-
Adaptation	-
Origination	-

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Demonstrate the closure properties of CFLs.
2. Draw a deterministic and non-deterministic finite automate which accept 00 and 11 at the end of a string containing 0, 1 in it, e.g., 01010100 but not 000111010.
3. State the difference between recursive and recursively enumerable language.
4. State the difference between PDA and TM.
5. Explain any two properties of recursive language in detail.

Course Outcome 2 (CO2)

1. Prove that the following languages are not regular using the pumping lemma.
$L = \{0^n 1^m 0^n | m, n \geq 0\}$
2. Prove that the following languages are not regular using the pumping lemma.
$L = \{wtw | w, t \in \{0,1\}^+\}$
3. Use the Pumping Lemma for context-free languages to show that the language is not context-free. $L = \{a^i b^j c^k | i < j < k\}$

Course Outcome 3 (CO3)

1. Implement a PDA accepting the language L ={ equal number of X‘s and Y‘s }by empty store.
2. Construct a PDA that accepts the language generated by grammar with productions S->aSbb | a.
3. Construct a TM for a language L ={an bn, n≥1}
4. Construct finite automata that accepting {11, 110}*{0}
5. Explain DFA with set of all strings such that that 10th symbol from the right end is ‘1‘.

Course Outcome 4 (CO4)

1. Construct CFG for the given PDA run for the string 000111
2. Construct CFG for the PDA that accepts the Dyck Language

3. Apply appropriate strength reduction techniques to eliminate unreachable non terminals in the produced CFG

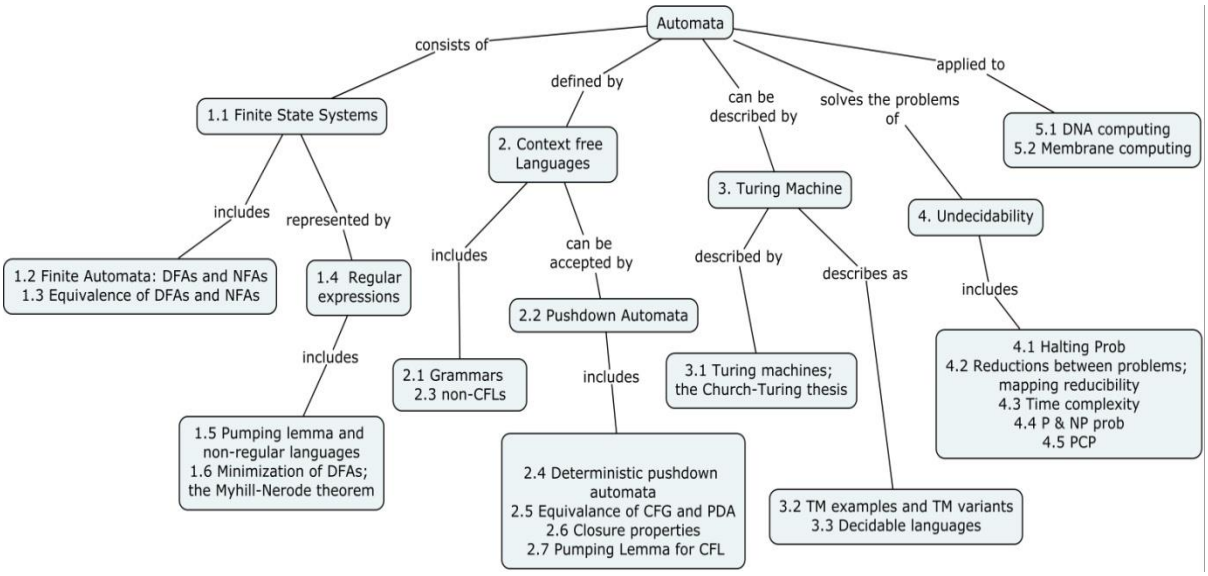
Course Outcome 5 (CO5):

1. Define Class P.
2. Define Class NP.
3. P belongs NP. Why?
4. Describe two classes of decision problem P and NP.
5. Explain that minimum spanning tree is in NP.

Course Outcome 6 (CO6):

1. Define Membrane Computing
2. List the various types of Membrane systems
3. Explain the intuition behind Membrane
4. How does DNA computation solve the seven –point Hamiltonian path problem

Concept Map



Syllabus

Introduction to Finite Automata -Introduction, Methods of Proofs- Finite Automata: DFAs and NFAs- Equivalence of DFAs and NFAs, closure of regular operations- Regular expressions and equivalence with finite automata- Pumping lemma and non-regular languages- Minimization of DFAs; the Myhill-Nerode theorem- Closure property of regular languages/Decision properties of CFLA or regular languages- **Context Free Languages-**Context-free languages: introduction- Pushdown automata; equivalence with CFL- Examples of non-CFLs; Context-sensitive

languages; the Chomsky hierarchy- Deterministic pushdown automata- Equivalence to CFG and Deterministic PDA- Non Deterministic PDA- Equivalence between PDA accepting by null and final state- Equivalence between PDA and CFG- Closure properties of CFL- Pumping lemma for CFL – problems - Recursive, Recursively Enumerable language, Grammar Reduction- **Turing Machines-** Turing machines; the Church-Turing thesis-TM examples and TM variants- Decidable languages-**Undecidability-**Diagonalization; The Halting Problem- Greibach NF and DCFLS- Reductions between problems; mapping reducibility- Time complexity; The class P- The classes NP and co-NP; Examples; Polynomial-time reductions- Post Correspondence Problem- Godel's incompleteness theorem - **Case Study: New Models of Computation-** DNA Computing- Membrane Computing- Quantum Computing

Learning Resources

1. Kamala Krithivasan, R Rama, Introduction to Formal Languages, Automata Theory and Computation, Pearson India, 2009
2. Peter Linz, An Introduction to Formal Languages and Automata, Fifth Edition, Jones & Bartlett Learning, 2012.
3. J.E. Hopcroft, R. Motwani and J.D. Ullman, —Introduction to Automata Theory, Languages and Computations, second Edition, Pearson Education, 2007.
4. MichealSipser, —Introduction of the Theory and Computation, Thomson, second Edition 2005.
5. H.R. Lewis and C.H. Papadimitriou, —Elements of the theory of Computation, Second Edition, Pearson Education, 2003.
6. Thomas A. Sudkamp, An Introduction to the Theory of Computer Science, Languages and Machines, Third Edition, Pearson Education, 2007.
7. Raymond Greenlaw an H.James Hoover, —Fundamentals of Theory of Computation, Principles and Practice, Morgan Kaufmann Publishers, 1998.
8. J. Martin, —Introduction to Languages and the Theory of computation, Third Edition, Tata Mc Graw Hill, 2007.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcomes
1	**Introduction to Finite Automata(7)**		
1.1	Introduction, Methods of Proofs	1	CO1
1.2	Finite Automata: DFAs and NFAs	1	CO1
1.3	Equivalence of DFAs and NFAs, closure of regular operations	1	CO1
1.4	Regular expressions and equivalence with finite automata	1	CO1
1.5	Pumping lemma and non-regular languages	1	CO2
1.6	Minimization of DFAs; the Myhill-Nerode theorem	2	CO2
2	**Context Free Languages(10)**		
2.1	Context-free languages: introduction	1	CO2
2.2	Pushdown automata; equivalence with CFL	1	CO2
2.3	Examples of non-CFLs; Context-sensitive languages; the Chomsky hierarchy	3	CO2
2.4	Deterministic Pushdown Automata, Non Deterministic PDA	1	CO3

2.5	Equivalence to CFG and Deterministic PDA- Non Deterministic PDA- Equivalence between PDA accepting by null and final state- Equivalence between PDA and CFG	2	CO3
2.6	Closure properties of CFL	1	CO3
2.7	Pumping lemma for CFL – problems- Recursive, Recursively Enumerable language, Grammar Reduction-	1	CO3
3	**Turing Machines(7)**		
3.1	Turing machines; the Church-Turing thesis	1	CO4
3.2	TM examples and TM variants	3	CO4
3.3	Decidable languages	3	CO4
4	**Undecidability(8)**		
4.1	Diagonalization; The Halting Problem - Greibach NF and I	3	CO5
4.2	Reductions between problems; mapping reducibility	1	CO5
4.3	Time complexity; The class P	1	CO5
4.4	The classes NP and co-NP; Examples; Polynomial-time reductions	2	CO5
4.5	Post Correspondence Problem - Godel's incompleteness theorem	1	CO5
5	**New Models of Computation(4)**		
5.1	DNA Computing	2	CO6
5.2	Membrane Computing, Quantum Computing	2	CO6
	Total	36	

Course Designer

1. Dr.K.Sundarakantham kskcse@tce.edu
2. Dr.R.Leena Sri rlsit@tce.edu

18CS530	DATA COMMUNICATION AND NETWORKS

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

This course on Computer Network provides an introduction to the basic concepts in networks, OSI and TCP/IP reference models, layers, protocols, switching, routing and various applications that use Computer Networks. The objective of this course is to introduce the concepts in Computer Networks. Emphasis will be given to different layers and functionality of the TCP/IP protocol suite. At the end of the course, the students should have an understanding of the basic principles and practice of Computer Networking.

Prerequisite

Digital Circuits

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Explain the operation of network applications with an understanding of the network models, switching techniques and layered architecture.	20
CO2	Illustrate the concepts for encoding, multiplexing and different modulation techniques.	10
CO3	Solve flow and error control issues in the data link layer, using appropriate techniques.	20
CO4	Identify the performance implications of random access protocols like ALOHA, slotted ALOHA , CSMA/CD and CSMA/CA.	10
CO5	Construct routing and forwarding solutions for packet switching networks, with an understanding of the router architectures, algorithms and protocols.	25
CO6	Identify the performance of transport layer protocols under given scenario.	15

*** Weightage depends on Bloom's Level, number of contact hours,

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS2	Understand	Respond	Guided Response	1.2
CO2	TPS2	Understand	Respond	Guided Response	1.2
CO3	TPS3	Apply	Value	Mechanism	1.2,2.1.1,3.1.1
CO4	TPS3	Apply	Value	Mechanism	1.2,2.1.1,3.1.1

CO5	TPS3	Apply	Value	Mechanism	1.2,2.1.1,3.1.1,4.4.1
CO6	TPS3	Apply	Value	Mechanism	1.2,2.1.1,3.1.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-										M		
CO2	M	L	-										M		
CO3	S	M	L		L			L	L	L		M	M		L
CO4	S	M	L		L			L	L	L		M	M		L
CO5	S	M	L		L			L	L	L		M	M		L
CO6	S	M	L		L			L	L	L		M	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	2	3	1	2	3	
Remember	30	20	20		-	-	20
Understand	50	40	20	30	30	30	20
Apply	20	40	60	70	70	70	60
Analyse							
Evaluate							
Create							

Assessment Pattern: Psychomotor

Psychomotor Skill	~~Miniproject~~/Assignment/~~Practical Component~~
Perception	
Set	
Guided Response	30
Mechanism	70
Complex Overt Responses	
Adaptation	
Origination	

Sample Questions for Course Outcome Assessment**

** (2 to 3 at the cognitive level of course outcome)

Course Outcome 1(CO1):

1. Why is it said that FTP sends control information “out-of-band”?(Understand)
2. Describe the general format of HTTP request message. (Understand)
3. Explain the operation of domain name resolution. (Understand)

Course Outcome 2(CO2):

1. State the different types of Encoding. (Remember)
2. What is the difference between AM and FM. (Understand)

3. Explain the different types of multiplexing. (Understand)

Course Outcome 3 (CO3):

1. Sixteen-bit messages are transmitted using a Hamming code, using even parity. Determine the number of check bits needed to ensure that the receiver can detect and correct single bit errors? Show the bit pattern transmitted for the message1101001100110101. (Apply)
2. A 12-bit Hamming code whose hexadecimal value is 0xE4F arrives at a receiver. Determine the original transmitted value in hexadecimal, assuming not more than 1 bit is in error. (Apply)
3. A bit stream 10011101 is transmitted using the standard CRC method, with the generator 1001. Show the actual bit string transmitted. Suppose the third bit from the left and the second bit from the right of the transmitted message are inverted during transmission. Show that this error is detected (Apply)

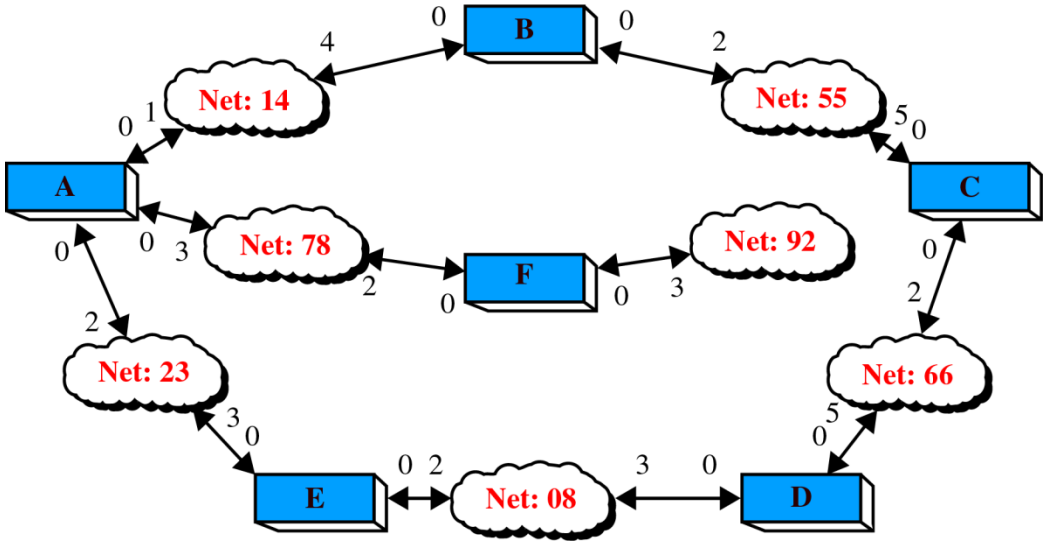
Course Outcome 4(CO4):

1. Suppose nodes A and B are on the same 10 Mbps broadcast channel, and the propagation delay between the two nodes is 245 bit times. Suppose A and B send Ethernet frames at the same time, the frames collide, and then A and B choose different values of *K* in the CSMA/CD algorithm. Assuming no other nodes are active, can the retransmissions from A and B collide? For our purposes, it suffices to work out the following example. Suppose A and B begin transmission at $t = 0$ bit times. They both detect collisions at $t = 245$ bit times. Suppose $KA = 0$ and $KB = 1$. At what time does B schedule its retransmission? At what time does A begin transmission? At what time does A's signal reach B? Does B refrain from transmitting at its scheduled time? (Apply)

2. Suppose four active nodes—nodes A, B, C and D—are competing for access to a channel using slotted ALOHA. Assume each node has an infinite number of packets to send. Each node attempts to transmit in each slot with probability *p*. The first slot is numbered slot 1, the second slot is numbered slot 2, and so on.
   a) What is the probability that node A succeeds for the first time in slot 5?
   b) What is the probability that some node (A, B, C or D) succeeds in slot 4?
   c) What is the probability that the first success occurs in slot 3?(Apply)

Course Outcome 5 (CO5):

1. Consider the following datagram network. Show the forwarding table in router A, such that all traffic destined to host H3 is forwarded through interface 3.Also write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4. (Apply)
2. An organization has granted a block of address with the beginning address 25.24.74.0/24.The organization need to have 3 subblocks to be used in 3 subnets: one subblock of 10 addresses, one subblock of 50 addresses and one subblock of 125 addresses. Design the subblocks.

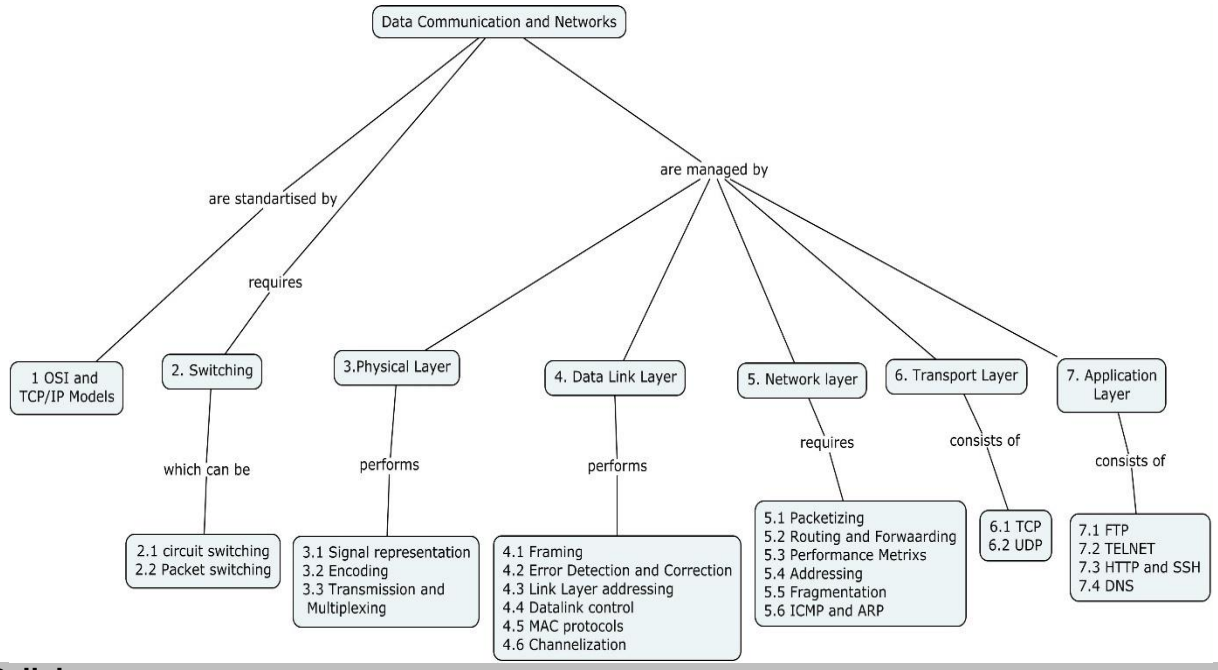


For the subnet shown in the figure calculate the shortest path for all the networks form **Router A** by using Dijkstra algorithm (Show intermediate steps) and build the routing table for **Router A.**

Course Outcome 6 (CO6):

1. Consider transferring an enormous file of *L* bytes from Host A to Host B. Assume an MSS of 536 bytes. What is the maximum value of *L* such that TCP sequence numbers are not exhausted? For the calculated value of L, find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and link header are added to each segment before the resulting packet is sent out over a 155 Mbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously. (Apply)
2. Suppose Host A sends two TCP segments to Host B over a TCP connection. The first segment has a sequence number 90; the second has a sequence number 110. How much data is in the first segment? If the first segment is lost but the second segment arrives at B. What will be the acknowledgment number in the acknowledgment that Host B sends to Host A.(Apply)

Concept Map



Syllabus

Network Models - The OSI Model - TCP/IP Protocol Suite - OSI versus TCP/IP – **Switching** - Circuit-Switched Networks - Packet Switching, Structure of a Switch, Switching and TCP/IP Layers -**Physical Layer**– Signal representation, Encoding ,Transmission and Multiplexing-**Data link Layer**: Framing, Error Detection and Correction: Types of Errors - Error Detection, Cyclic Redundancy Check, Checksum, Forward Error Correction, Link-Layer Addressing - Data link Control - Media Access Control (MAC): CSMA/CD, CSMA/CA, Controlled Access – Channelization: FDMA, TDMA, CDMA -**Network Layer** - Connecting Devices: Hubs, Link-Layer Switches, Routers – Packetizing - Routing and Forwarding: Distance-Vector Routing, Link-State Routing - Performance Metrics: Delay, Throughput, Packet Loss, Congestion Control - Addressing: Internet Protocol, IPV6,IPV4 Addresses: Classful Addressing, Classless Addressing - Fragmentation – Internet Control Message Protocol (ICMP) - Address Resolution Protocol (ARP)- RARP -**Transport Layer**: Transmission Control Protocol, User Datagram Protocol- **Application Layer:** FTP - TELNET HTTP- Secure Shell (SSH) - Domain Name System (DNS)

Learning Resources

1. Data Communications and Networking, 5th Edition, BehrouzForouzan, Mc Graw Hill, 2017
2. Computer Networks: A Systems Approach, Larry L. Peterson, Bruce S. Davie, Elsevier, Mar 2011
3. Computer Networking: A Top-Down Approach featuring the Internet, 6th edition, James F. Kurose, Pearson Education India, 2013.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
1.	**Network Models** - The OSI Model - TCP/IP Protocol Suite - OSI versus TCP/IP	1	CO1
2	**Switching**		
2.1	Circuit-Switched Networks	1	CO1

2,2	Packet Switching, Structure of a Switch, Switching and TCP/IP Layers	1	CO1
3	**Physical Layer**		
3.1	Signal representation	1	CO2
3.2	Encoding	2	CO2
3.3	Transmission and Multiplexing	2	CO2
4	**Data link Layer**		
4.1	Framing	1	CO3
4.2	Error Detection and Correction - Types of Errors, Error Detection, Cyclic Redundancy Check, Checksum, Forward Error Correction	3	CO3
4.3	Link-Layer Addressing	1	CO4
4.4	Data link Control	1	CO4
4.5	Media Access Control (MAC): CSMA/CD(Ethernet), CSMA/CA, Controlled Access	2	CO4
4.6	Channelization: FDMA, TDMA, CDMA	1	CO4
5	**Network Layer** - Connecting Devices: Hubs, Link-Layer Switches, Routers	2	CO5
5.1	Packetizing	1	CO5
5.2	Routing and Forwarding-Distance-Vector and Link-State Routing	2	CO5
5.3	Performance Metrics : Delay, Throughput, Packet Loss, Congestion Control	1	CO5
5.4	Addressing: Internet Protocol –IPV6, IPV4 Addresses: Classful Addressing, Classless Addressing	2	CO5
5.5	Fragmentation	1	CO5
5.6	Internet Control Message Protocol(ICMP), Address Resolution Protocol (ARP), RARP	1	CO5
6	**Transport Layer**	1	CO6
6.1	Transmission Control Protocol	3	CO6
6.2	User Datagram Protocol	1	CO6
7	**Application Layer**		
7.1	FTP	1	CO1
7.2	TELNET	1	CO1
7.3	HTTP,Secure Shell (SSH)	1	CO1
7.4	Domain Name System (DNS)	1	CO1
	Total	36	

Course Designers:

1. Dr.C.Senthilkumar cskcse@tce.edu
2. Dr.G.S.R.EmilSelvan emil@tce.edu

		Category	L	T	P	Credit
18CS560	**SOFTWARE ENGINEERING: THEORY AND PRACTICE**	PC	2	0	2	3

Preamble

The main purpose of this course is to impart knowledge on various models (interaction, context models etc.) and processes that are used by professionals in the field of software engineering. This course focuses on architecture patterns and various software engineering methodologies for designing and developing the software. Consequently, student's take up a group project, working through a number of stages for the development of software.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course, students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Explain various software development process and management techniques	10
CO2	Understand various types of software like client server, fault tolerant etc.	20
CO3	Design the model for the given software requirements using Context, Interaction models etc	20
CO4	Analyze the key techniques involved in testing the software based on the requirements.	20
CO5	Develop the system partially through Test Driven Development.	10
CO6	Build a project report as a team which contains the requirement specification, plan, schedule and design documents	20

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS2	Understand	Respond	Guided Response	1.2, 4.1.1, 4.1.2
CO2	TPS2	Understand	Respond	Guided Response	1.2, 4.1.1, 4.1.2
CO3	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.5, 4.1.1, 4.1.2, 4.4.1, 4.4.3
CO4	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 4.1.1, 4.1.2
CO5	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 3.1.1, 4.1.1, 4.1.2
CO6	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 3.1.1, 4.1.1, 4.1.2

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L									L		L		L
CO2	M	L									L		L		L
CO3	S	M	L	L	S	S	M	M			L	M	M	S	M
CO4	S	M	L	L	S	S	M	M			L	M	M	S	M
CO5	S	M	L	L	S	S	S	S	S	M	M	M	M	S	M
CO6	S	M	L	L	S	S	S	S	S	M	M	M	M	S	M

S- Strong; M-Medium; L-Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests				Terminal Examination
	1	2	3	3(Practical)	Theory
Remember	20	20	20	-	20
Understand	30	30	30	-	30
Apply	50	50	50	100(Team work)	50
Analyse	-	-	-	-	-
Evaluate	-	-	-	-	-
Create	-	-	-	-	-

Assessment Pattern: Psychomotor

Psychomotor Skill	**Miniproject/Assignment/Practical Component**
Perception	
Set	
Guided Response	30
Mechanism	70
Complex Overt Responses	
Adaptation	
Origination	

Sample Questions for Course Outcome Assessment**

** (2 to 3 at the cognitive level of course outcome)

Course Outcome 1 (CO1):

1. Explain how electronic connectivity between various development teams can support software engineering activities. (Understand)
2. Why it is important to make a distinction between developing the user requirements and developing system requirements in the requirements engineering process. (Understand)

3. Explain how the principles underlying agile methods lead to the accelerated development and deployment of software. (Understand)

Course Outcome 2 (CO2):

1. What are the essential differences between CBSE with reuse and software processes for original software development? (Understand)
2. Your customer wants to develop a system for stock information where dealers can access information about companies and evaluate various investment scenarios using a simulation system. Each dealer uses this simulation in a different way, according to his or her experience and the type of stocks in question. Suggest a client–server architecture for this system that shows where functionality is located. Justify the client–server system model that you have chosen. (Understand)
3. Giving reasons for your answer, suggest two important types of application where you would not recommend the use of service-oriented architecture. (Understand)

Course Outcome 3 (CO3):

1. Suggest how an engineer responsible for drawing up a system requirements specification might keep track of the relationships between functional and non-functional requirements. (Apply)
2. Draw a sequence diagram for the same system. Explain why you might want to develop both activity and sequence diagrams when modeling the behavior of a system. (Apply)
3. Suggest an architecture for a system (such as iTunes) that is used to sell and distribute music on the Internet. What Architectural patterns are the basis for your proposed architecture? (Apply)

Course Outcome 4 (CO4):

1. What are the benefits of involving users in release testing at an early stage in the testing process? Are there disadvantages in user involvement? (Understand)
2. Explain why legacy systems should be thought of as socio technical systems rather than simply software systems that were developed using old technology. (Understand)
3. The reuse of software raises a number of copyright and intellectual property issues. If a customer pays a software contractor to develop a system, who has the right to reuse the developed code? Does the software contractor have the right to use that code as a basis for a generic component? What payment mechanisms might be used to reimburse providers of reusable components? Discuss these issues and other ethical issues associated with the reuse of software. (Apply)

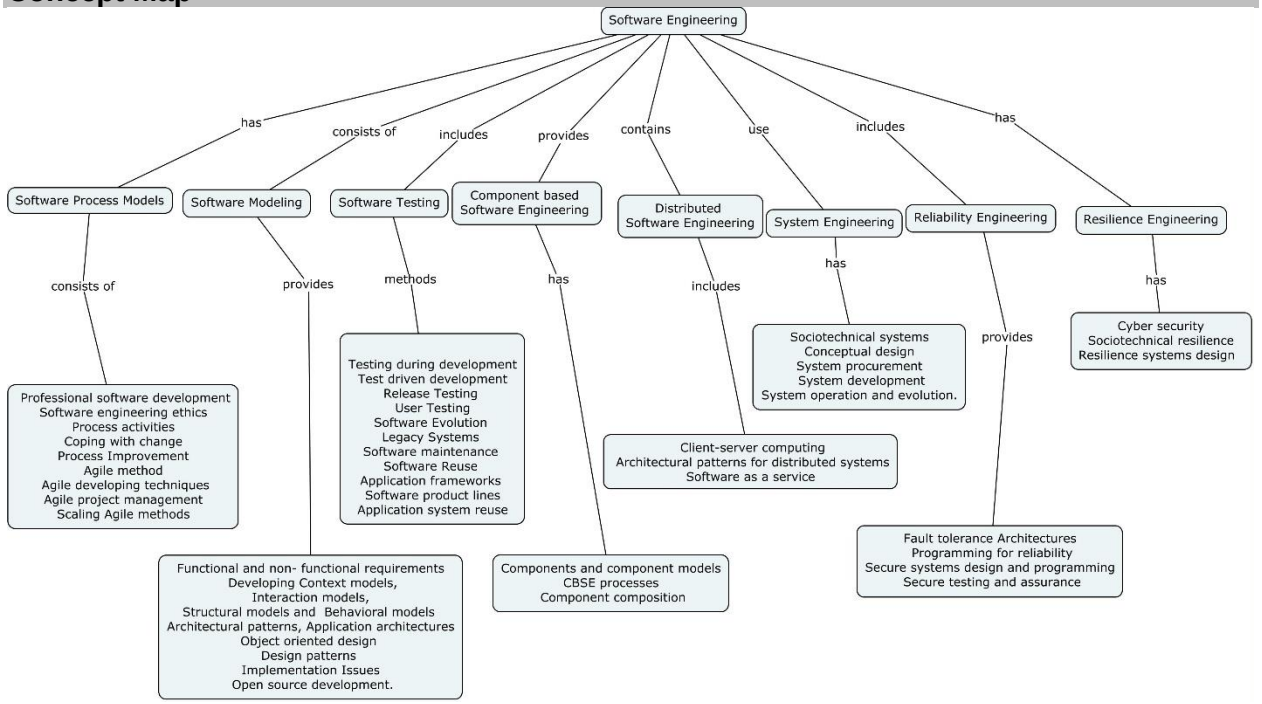
Course Outcome 5 (CO5):

1. Design Graphical user interface for taken case study (Apply)
2. Develop acceptance test cases for selected case study. (Apply)
3. Develop the code for selected case study (Apply)

Course Outcome 6 (CO6):

1. Select software life cycle model suitable for case study and justify your answer. (Apply)
2. Develop user stories and story map for case study. (Apply)
3. Construct a design diagrams for case study. (Apply)

Concept Map



Syllabus

Software process models

Professional software development- Software engineering ethics- Process activities- Coping with change- Process Improvement- Software Life cycle- Iterative, Spiral, Prototyping-Agile method- Agile developing techniques- Agile project management- Scaling Agile methods.

System Modelling

Functional and non- functional requirements- Software Requirement Specification - Developing Context models, Interaction models, Structural models and Behavioural models- Architectural patterns, Application architectures- -Object oriented design- Design patterns- Implementation Issues- Open source development.

Software Testing

Testing during development- Test Driven Development- Release Testing- User Testing- Software Evolution- Legacy Systems- Software maintenance- Software Reuse- Risk Management

Component based Software Engineering

Components and component models- CBSE processes- Component composition

Distributed Software Engineering

Distributed systems- Client-server computing- Architectural patterns for distributed systems- Software as a service

System Engineering

Socio technical systems- Conceptual design- System procurement- System development- System operation and evolution.

Reliability Engineering

Fault tolerance Architectures- Programming for reliability- Secure systems design and programming- Secure testing and assurance-.

Resilience Engineering

Cyber security- Socio technical resilience- Resilience systems design.

Lab Content:

Develop a mini project for a real world problem in which a software solution can be obtained (a team of 3 members) and do the following.

- Collect requirements for the chosen problem
- Model the system through interaction, structural diagrams and develop software architecture.
- Develop the system partially through Test Driven Development with unit test.

Sample case-studies:

- An embedded control system for a personal insulin pump

This case study discusses the control software for a personal insulin pump, which is used by diabetics to mimic the function of the pancreas and hence control the level of glucose (sugar) in their blood.

- The iLearn digital learning environment

The iLearn system is a digital learning environment used to support learning in schools with students from age 4 to 18. It is intended to replace an existing system (Glow) that was specially built for the purpose and which includes its own applications for e-mail, etc.

- The Mentcare system

This case study focuses on the requirements for a system that I have called the Mentcare system, which is a real system (although that is not its real name) which was used in a number of UK hospitals, including hospitals in Scotland.

- Wilderness weather station

This case study is based on the software for a wilderness weather station that collects weather information in remote areas that do not have local infrastructure (power, communications, roads, etc.).

Reference Books

1. Ian Sommerville , "Software Engineering" , 10th Edition,John Wiley and sons,2015.
2. Orit Hazzan, Yael Dubinsky, “Agile software engineering”, Springer,2014
3. The Unified Modeling Language Reference Manual, James Rumbaugh, Ivar Jacobson, Grady Booch, 2nd Edition, Addison Wesley,2005.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	**Software process models**		
1.1	Professional software development- Software engineering ethics- Process activities- Coping with change- Process Improvement	1	CO1
1.2	Software Life cycle models- Iterative, Spiral and Prototyping models	1	CO1
1.3	Agile method- Agile developing techniques- Agile project management- Scaling Agile methods	1	CO1
2	**System Modelling**		
2.1	Functional and non- functional requirements- SRS	1	CO3
2.2	Developing Context models, Interaction models	1	CO3
2.3	Architectural patterns, Application architectures	1	CO3
2.4	Structural models and Behavioural models	1	CO3
2.5	Object oriented design	1	CO3
2.6	Design patterns- Implementation Issues- Open source development	1	CO3
3	**Software Testing**		
3.1	Testing during development- Test driven development- Release Testing- User Testing	1	CO4
3.2	Software Evolution- Legacy Systems- Software maintenance, Software Reuse- Software product lines- Application system reuse	1	CO4
3.3	Risk Management	1	CO1
4	**Component based Software Engineering**		
4.1	Components and component models	1	CO2
4.2	CBSE processes- Component composition	1	CO2
5	**Distributed Software Engineering**		
5.1	Distributed systems- Client-server computing-	1	CO2

Module No.	Topic	No. of Lectures	Course Outcome
1	**Software process models**		
5.2	Architectural patterns for distributed systems- Software as a service	1	CO2
6	**System Engineering**		
6.1	Socio technical systems- Conceptual design	1	CO2
6.2	System procurement- System development	1	CO2
6.3	System operation and evolution.	1	CO2
7	**Reliability Engineering**		
7.1	Fault tolerance Architectures- Programming for reliability	1	CO2
7.2	Secure systems design and programming	1	CO2
7.3	Secure testing and assurance	1	CO2
8	**Resilience Engineering**		
8.1	Cyber security- Socio technical resilience	1	CO2
8.2	Resilience systems design	1	CO2

Module No.	Topic	No. of Lectures		
1	Develop a mini project for a real world problem in which a software solution can be obtained (a team of 3 members) and do the following, Collect requirements for the chosen problem	4	CO6	
2	Develop context level models and software architecture.	4	CO6	
3	Model the system through structural diagrams	6	CO6	
4	Model the system through interaction diagrams	6	CO6	
5	Develop the system partially through Test Driven Development with unit test.	4	CO5	

Course Designers:

1. Dr. A. Malini amcse@tce.edu
2. Mr. V. Vignaraj Ananth vignaraj@tce.edu

18CS570	DATABASES LAB

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

This course aims at facilitating the student to apply the effective designing of relational database for Real-world applications, perform many operations related to creating, manipulating and maintaining databases using DBMS tools and manipulate data using the higher level language - JDBC.

Prerequisite

Database Management Systems

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Analyze and construct the Entity Relationship and Relational models for the given real-world application	10
CO2	Develop normalized database for given application using various constraints.	10
CO3	Build relational database and manipulate the same using simple and complex queries in SQL	20
CO4	Construct and make use of the database objects like Index, View, Sequence, Varray and Nested table using SQL	10
CO5	Develop database objects like Procedure, Functions, Triggers and Package using Pl/SQL and manipulate the database through these objects.	25
CO6	Develop a complete database application using higher level language through JDBC	25

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5
CO2	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5
CO3	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5
CO4	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5

CO5	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5
CO6	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 2.2.3, 2.4.3, 3.1, 4.4.3, 4.4.4, 4.5

Mapping with Programme Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		L	M	M		L	M	L	M
CO2	S	M	L		M	L		L	M	M		L	M	L	M
CO3	S	M	L		S	L		L	M	M		L	M	M	M
CO4	S	M	L		M			L	M	M			M	M	M
CO5	S	M	L		S	L		L	M	M			M	M	M
CO6	S	M	L		S	L		L	M	M		L	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember	-	-
Understand	-	-
Apply	100	100
Analyse	-	-
Evaluate	-	-
Create	-	-

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Practical Component/Observation
Perception	-
Set	-
Guided Response	25
Mechanism	75
Complex Overt Responses	-
Adaptation	-
Origination	-

List of Experiments/Activities with CO Mapping

Module No.	Topic	No. of Sessions	Course Outcome
1.	Analyze and model a database application	2	CO1
2.	Creation and Modification of tables with and without normalization	2	CO1, CO2
3.	Complete database creation using Integrity Constraints and giving privilege on the database to other users	2	CO2
4.	Manipulating the database using simple SQL Queries and transaction control using COMMIT, ROLLBACK and SAVEPOINT commands	2	CO3
5.	Manipulating the database using Complex SQL Queries	2	CO3
6.	Creation and usage of database objects, complex data types such as BLOB, CLOB, NCLOB, BFILE	2	CO4
7.	Creation of Functions, Procedures with cursors and exception handling using PL/SQL	2	CO5
8.	Creation of Simple and Compound Triggers using PL/SQL	2	CO5
9.	Creation of Package using Pl/SQL	2	CO5
10.	Database application using JDBC	2	CO6
11.	Demonstration of a complete database application using PL/SQL or JDBC as front end – Report submission	4	CO1..CO6
	Total	**24**	

Learning Resources

1. https://courses.tce.edu/
2. https://apex.oracle.com/en/

Course Designers:

1. Mrs. A.M.Rajeswari amrcse@tce.edu
2. Mrs.B.Subbulakshmi bscse@tce.edu

18CS580	NETWORK PROGRAMMING LAB

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

Universal connectivity is realized through Computer Networks. It is important to gain knowledge on the hardware requirements and functioning of Computer Networks. This course provides insight into the working of network protocols and their characteristics.

Prerequisite

Object oriented programming

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Perform Configuration of networking components and installing device drivers and build a Local Area Network.	10
CO2	Perform port scanning and identify IP and MAC Address.	20
CO3	Implement client server communication using socket programming and Applet.	25
CO4	Perform DNS server host name identification and resolve given host name	10
CO5	Implement File transfer and RMI.	25
C06	Simulate a network topology using NS3.	10

*** Weightage depends on Bloom's Level, number of contact hours,

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2,3.2.3,4.5.1
CO2	TPS3	Apply	Value	Mechanism	1.2,3.2.3,4.5.1,4.5.3
CO3	TPS3	Apply	Value	Mechanism	1.2,3.2.3,4.5.1,4.5.3
CO4	TPS3	Apply	Value	Mechanism	1.2,3.2.3,4.5.1,4.5.3
CO5	TPS3	Apply	Value	Mechanism	1.2,3.2.3,4.5.1,4.5.3
C06	TPS3	Apply	Value	Mechanism	1.2,2.1.2,3.2.3,4.5.1,4.5.3

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L			L	L	M	L	L		L	M	L	L
CO2	S	M	L		L	L	L	M	L	L		L	M	L	L
CO3	S	M	L		L	L	L	M	L	L		L	M	L	L

CO4	S	M	L		L	L	L	M	L	L		L	M	L	L
CO5	S	M	L		L	L	L	M	L	L		L	M	L	L
CO6	S	M	L		L	L	L	M	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand	20	20
Apply	80	80
Analyse		
Evaluate		
Create		

Assessment Pattern: Psychomotor

Psychomotor Skill	~~Miniproject /~~Practical Component/~~Observation~~
Perception	
Set	
Guided Response	20
Mechanism	80
Complex Overt Responses	
Adaptation	
Origination	

List of Experiments/Activities with CO Mapping

Module No.	Topic	No. of Sessions	Course Outcome
1	Establishment of a LAN: Preparation of network cables and installation and configuration of network.	2	CO1
2	Write a program to identify your machine's host name and IP address.	2	CO2
3	Write a program to locate the next hop router's IP address and MAC address.	2	CO2
4	Write a program to find which port is currently used, by scanning the port.	2	C02
5	Write a program to obtain local DNS server's host name and IP address and resolve a given host name.	2	CO3
6	Write a program to illustrate a simple client/server communication and Time server	2	CO4
7	Write a program to implement ECHO and PING commands and time server.	2	CO4
8	Develop a client server application for chat using Applets	2	C04
9	Write a program to implement a file transfer using TCP.	2	CO5

Module No.	Topic	No. of Sessions	Course Outcome
10	Write a program to implement a file transfer using UDP.	2	CO5
11	Write a program to implement Remote Method Invocation.	2	CO5
12	Simulate a network topology using NS3.	2	C06
Total		24	

Course Designers:

1. Dr.C.Senthilkumar cskcse@tce.edu
2. Dr.G.S.R.EmilSelvan emil@tce.edu

18ES590	SYSTEM THINKING

Category	L	T	P	Credit
ES	1	-	2	2

Preamble

Systems thinking is the integrated paradigm for systems science and system approaches to practice. It is concerned with understanding or intervening in problem situations, based on the principles and concepts of the system model. It can help to provide a common language and an intellectual foundation and make practical system concepts, principles, patterns and tools accessible to systems engineering. System thinking considers the similarities between systems from different domains in terms of a set of common systems concepts, principles, and patterns. The scope of systems thinking is a starting point for dealing with real-world situations using a set of related systems concept. The system thinking is viewed as both a set of founding ideas for the development of systems theories and practices and also as a pervasive way of thinking need by those developing and applying them. This systems approach is a way of tackling real-world problems and making use of the concepts, principle, patterns of systems thinking to enable the systems to be engineered and used.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage in %
CO1	Explain the concepts of systems thinking, System engineering and Systems Life Cycle	10
CO2	Identify system elements, interactions, boundary and environment for the given system descriptions	20
CO3	Develop a functional architecture with appropriate primary function(s) and sub-functions of the identified system	10
CO4	Develop a physical architecture with appropriate sub-systems and components of the identified system	10
CO5	Prepare a system requirement specification review documents for the various stages of acquisition phase of the identified system	20
CO6	Develop a system model with logical and physical architecture using system modelling tool like SysML	30

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS2	Understand	Respond	-	1.1, 2.3.1, 2.3.2
CO2	TPS3	Apply	Value	-	1.1, 2.1.1, 2.3.1,2.3.2, 2.3.3, 2.3.4, 2.4.4, 4.3.1,
CO3	TPS3	Apply	Value	Mechanism	1.1, 2.1.1, 2.3.1,2.3.2, 2.3.3, 2.3.4, 2.4.4, 3.1.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6, 4.3.1, 4.3.2, 4.3.3, 4.4.5, 4.5.1
CO4	TPS3	Apply	Value	Mechanism	1.1, 2.1.1, 2.3.1,2.3.2, 2.3.3, 2.3.4, 2.4.4, 3.1.2, 3.2.3, 3.2.4,

					3.2.5, 3.2.6, 4.3.1, 4.3.2, 4.3.3, 4.4.5, 4.5.1
CO5	TPS3	Apply	Value	Mechanism	1.1, 2.1.1, 2.3.1,2.3.2, 2.3.3, 2.3.4, 2.4.4, 3.1.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6, 4.3.1, 4.3.2, 4.3.3, 4.4.5, 4.5.1
CO6	TPS3	Apply	Value	Mechanism	1.1, 2.1.1, 2.3.1,2.3.2, 2.3.3, 2.3.4, 2.4.4, 3.1.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6, 4.3.1, 4.3.2, 4.3.3, 4.4.5, 4.5.1

Mapping with Programme Outcomes and Programme Specific Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-	-	-	-	-	-	-	-	-	-			
CO2	S	M	L	-	-	L	L	L	L	L	-	M			
CO3	S	M	L	-	-	M	M	M	L	M	M	S			
CO4	S	M	L	-	-	M	M	M	L	M	M	S			
CO5	S	M	L	-	-	M	M	M	L	M	M	S			
CO6	S	M	L	-	S	M	M	M	L	M	M	S			

S- Strong; M-Medium; L-Low

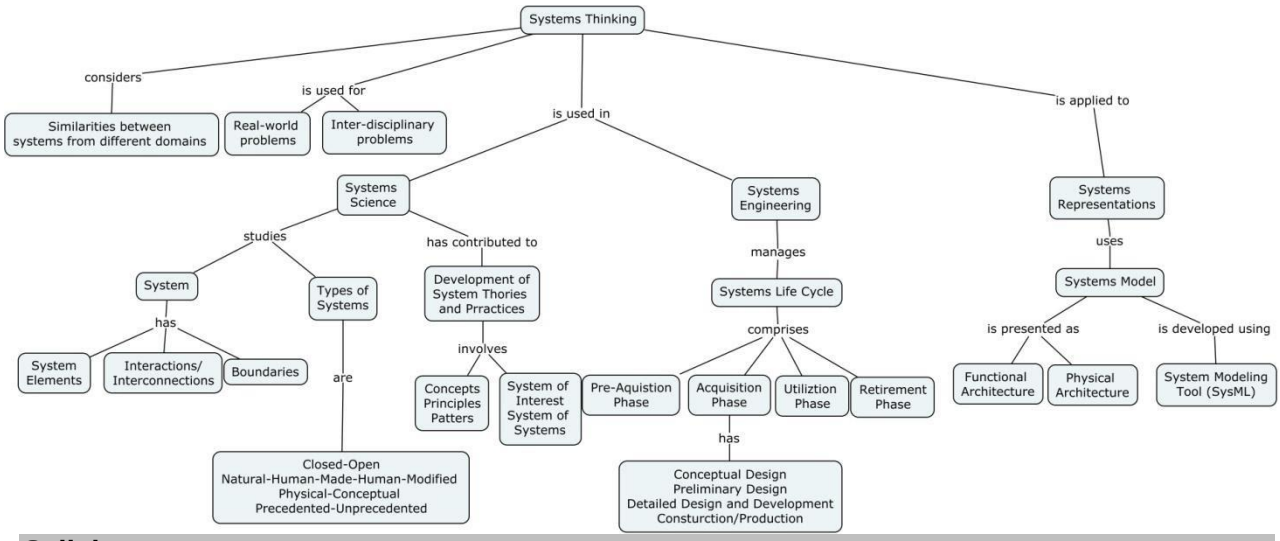
Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Test -2
Remember	20
Understand	40
Apply	40
Analyse	-
Evaluate	-
Create	-

Phases	Deliverables	Marks	Course Outcomes
Continuous Assessment			
Continuous Assessment Test-2	Summative Assessment Report	10	CO1 and CO2
Review 1 – Functional & Physical Architecture and System Requirement Specification	Technical Report	25	CO3, CO4 and CO5
Review 2 – Systems Modeling	Technical Report	15	CO6
End-Semester Examination			
Demonstration	Virtual Prototype with simulation	60	CO1, CO2, CO3, CO4 CO5 and CO6
Poster Presentation	Poster	40	

- Reports are to be submitted at each review. The report and presentation will be evaluated based on Rubrics
- Demonstration of Virtual Prototype with simulation and Poster presentation will be evaluated by two faculty members nominated by their respective Head of the Department.

Concept Map



Syllabus

1.0 Systems Fundamentals: System - Definition, System Elements, Interactions, System Boundary, - Types of Systems: Closed-Open, Natural-Human-Made-Human-Modified, Physical-Conceptual and Precedented-Unprecedented. Systems science - Systems approaches. Systems Thinking: Concepts, principles and pattens. System of Interest - Systems of System. Systems Engineering: Product, Service, Enterprise. System Life Cycle: Pre-acquisition phase, Acquisition Phase, Utilization Phase and Retirement Phase.

2.0 Acquisition Phase: Conceptual Design: Business needs and requirements, Stakeholder needs and requirements, System Requirement Specification, Functional Base Line, System Requirement Review – Functional Architecture. Preliminary Design: Configuration items, Allocated Baseline, Preliminary Design Review – Physical Architecture. Detailed Design and Development: System Modeling, Product Base Line, Critical Design Review. Construction/Production: Formal Qualification Review, Acceptance Test and Evaluation.

3.0 Systems Modeling: System Model - Types of models – System Modeling Concepts – Modeling Standards. System Architecture: Logical Architecture Model – Physical Architecture Model. Systems Life Cycle Process Model: Vee model.

Learning Resources

1. A Guide to Guide to the Systems Engineering Body of Knowledge (SEBoK), version 2.2, INCOSE Systems Engineering Research Center and IEEE Computer Society, Released 31 October 2019 – https://www.sebokwiki.org/w/images/sebokwiki-farm!w/8/8b/SEBoK_v2.1.pdf
2. Systems Engineering Handbook, A Guide for Systems Life Cycle Processes and Activities, 4th Edition, INCOSE-TP-2003-002-04, 2015.
3. R. Ian Faulconbridge, Michael Ryan, “Systems Engineering Practice”, Argos Argos Press, 2014.
4. Jon Holt and Simon Perry, “SysML for systems engineering”, The Institution of Engineering and Technology, London, United Kingdom, 2008.

5. Sanford Friedenthal, Alan Moore and Rick Steiner, "A Practical Guide To SysML: The Systems Modeling Language, Third edition, Morgan Kaufmann, an imprint of Elsevier, 2015
6. Coursera course on Introduction to Systems Engineering - R. Ian Faulconbridge, Michael Ryan of The University of New South Wales, Sydney.
7. NPTEL Course: Systems Engineering Theory and Practice – IIT Kanpur – Prof. Deepu Philip (Last offered in 2019) - https://nptel.ac.in/courses/110/104/110104074/

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours		Course Outcome
		In-Class	Hands-on	
1.	**1.0 Systems Fundamentals:** System - Definition, System Elements, Interactions, System Boundary	1	-	CO1
1.1	Types of Systems: Closed-Open, Natural-Human-Made-Human-Modified, Physical-Conceptual and Precedented-Unprecedented.	1	2	CO1
1.2	Systems science - Systems approaches.	1	-	CO1
1.3	Systems Thinking: Concepts, principles and pattens.	1	-	CO1
1.4	System of Interest - Systems of System. Systems Engineering: Product, Service, Enterprise System Life Cycle: Pre-acquisition phase, Acquisition Phase, Utilization Phase and Retirement Phase.	2	2	CO2
2.	**Acquisition Phase**			
2.1	Conceptual Design: Business needs and requirements, Stakeholder needs and requirements, System Requirement Specification, Functional Base Line, System Requirement Review – Functional Architecture.	1	4	CO3
2.2	Preliminary Design: Configuration items, Allocated Baseline, Preliminary Design Review – Physical Architecture.	1	4	CO3
2.3	Detailed Design and Development: System Modeling, Product Base Line, Critical Design Review.	1	4	CO4
2.4	Construction/Production: Formal Qualification Review, Acceptance Test and Evaluation.	1	2	CO5
3.	**Systems Modelling**			
3.1	System Model - Types of models – System Modeling Concepts – Modeling Standards.	1	2	CO6
3.2	System Architecture: Logical Architecture Model – Physical Architecture Model.	1	6	CO6
3.3	Systems Life Cycle Process Model: Vee model.	1	2	CO6
	Total	14	28	

Course Designers:

1. Dr.S.J.Thiruvengadam sjtece@tce.edu
2. Dr.S.Saravana Perumaal sspmech@tce.edu
3. Dr.C.Jeyamala jeyamala@tce.edu

18CHAC0	ESSENCE OF INDIAN KNOWLEDGE

Category	L	T	P	Credit
AC	2	0	0	0

Preamble

On the successful completion of the course, the students will be able to explain the concept of Indian Traditional Knowledge along with Indian Modern Knowledge. Traditional Knowledge Systems or Indigenous Knowledge Systems are a body of knowledge, which is very ancient and deep rooted. They have their origins in the remote past. Their systematisation and canonisation gave rise to the elite (the Greater Tradition) science. The nature of Traditional Knowledge System is diverse. It covers, among other things, literary, artistic and scientific works; songs, dances, medical treatments and practices; manufacturing and industry; and agricultural technologies and techniques. There is a dramatically growing national and international interest in incorporating Traditional Knowledge Systems, including Traditional Ecological Knowledge, into truly participatory approaches to development.

Course Outcome:

On the successful completion of the course students will be able to

CO1	Explain the concept of Traditional Knowledge and Modern knowledge of India.	Understand
CO2	Explain the need and importance of protecting Traditional Knowledge, Knowledge sharing, and Intellectual property rights over Traditional Knowledge.	Understand
CO3	Explain about the use of Traditional Knowledge to meet the basic needs of human being.	Understand
CO4	Explain the rich biodiversity materials and knowledge preserved for practicing traditional lifestyle.	Understand
CO5	Explain the use of Traditional Knowledge in Manufacturing and Industry.	Understand
CO6	Explain about the cultural expression and modern applications of Traditional Knowledge	Understand

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M
CO2	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M
CO3	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M
CO4	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M
CO5	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M
CO6	M	L	-	-	-	S	M	M	M	M	-	L	M	-	M

S- Strong; M-Medium; L-Low

Syllabus

Traditional and Modern Knowledge: Two Worlds of Knowledge - Phase of Explorers, Sir Arthur Cotton and Irrigation, Smallpox Vaccination, Late Nineteenth Century, Voelcker, Howard and Agriculture, Havell and Indian Art; Indians at the Encounter - Gaekwad of Baroda and Technical Education, Science Education and Modern Industries, Hakim Ajmal Khan and Ayurveda, R. N. Chopra and Indigenous Drugs, Gauhar Jaan and Indian Classical Music; Linking Science and the Rural - Tagore's Sriniketan Experiment, Marthandam, the YMCA Model, Gandhi's Thoughts on

Development, Nehru's View of Growth; Post-Independence Era - Modernization and Traditional Knowledge, Social Roots of Traditional Knowledge Activism, Global Recognition for Traditional Knowledge. **Global Mechanisms of Protection and Sharing:** For Recognition and Protection - United Nations Educational, Scientific and Cultural Organization (UNESCO), World Health Organization (WHO), International Labour Organization (ILO), UN Working Group on Indigenous Populations, Evolution of Other Organizations; Norms of Sharing - United Nations Environment Programme (UNEP), World Intellectual Property Organization (WIPO), World Trade Organization (WTO); IPR and Traditional Knowledge - Theoretical Background, Positive Protections of TK, Defensive Strategies, IPR Facilitation for TK. **Traditional Knowledge for Basic Needs:** Indian Midwifery Tradition—The Dai System, Surface Flow Irrigation Tanks, Housing - A Human Right, Changing Priorities—Niyamgiri. **Biodiversity and Genetic Resources:** Jeevani - The Wonder Herb of Kanis, A Holistic Approach - FRLHT, Basmati - In the New Millennium, AYUSH-Based Cosmetics. **Traditional Knowledge in Manufacturing and Industry:** Drug Discovery, A Sweetener of Bengal, The Sacred Ring of Payyanur, Channapatna Toys. **Traditional Cultural Expressions:** Banarasi Saree, Music, Built and Tangible Heritage, Modern Yoga, Sanskrit and Artificial Intelligence, Climate Change and Traditional Knowledge.

Assessment Pattern

Bloom's category	Continuous Assessment Tests		Seminar
	1	2	-
Remember	40	40	0
Understand	60	60	100
Apply	0	0	0
Analyze	0	0	0
Evaluate	0	0	0
Create	0	0	0

Learning Resources:

1. Nirmal Sengupta "Traditional Knowledge in Modern India Preservation, Promotion, Ethical Access and Benefit Sharing Mechanisms" Springer, 2019.
2. Amit Jha,"Traditional Knowledge System in India", Atlantic Publishers and Distributors Pvt Ltd, 2009.
3. Basanta Kumar Mohanta, Vipin Kumar Singh "Traditional Knowledge System and Technology in India", Pratibha Prakashan, 2012.
4. Kapil Kapoor, Michel Danino "Knowledge Traditions and Practices of India", Central Board of Secondary Education, 2012.
5. NPTEL video lecture on "Ayurvedic Inheritance of India", Video link: https://nptel.ac.in/courses/121/106/121106003/#.
6. Youtube video on "Introduction to Indian Knowledge Systems", Video link: https://www.youtube.com/watch?v=LZP1StpYEPM.
7. Youtube video on "12 Great achievements of Indian Civilization", Video link: https://www.youtube.com/watch?v=xmogKGCmclE.

Course Designers:

1. Dr.S.J.Thiruvengadam sjtece@tce.edu
2. Dr.V.R.Venkatasubramani venthiru@tce.edu