

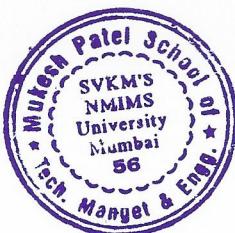
**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

<b>Program:</b> B. Tech. Computer Science & Business Systems				<b>Semester :</b> III	
<b>Course/Module:</b> Formal Language and Automata Theory				<b>Module Code:</b> BTCS03001	
<b>Teaching Scheme</b>				<b>Evaluation Scheme</b>	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks 50)	Term End Examinations (TEE) (Marks- 100 in Question Paper)
3	2	1	5	Marks Scaled to 50	Marks Scaled to 50
<b>Pre-requisite:</b> Discrete Mathematics (BTCS01001), Data Structures and Algorithms (BTCS02003)					
<b>Objectives:</b> To introduce fundamentals of automata theory, regular sets and grammars, To design automata for different types of languages					
<b>Outcomes:</b> After completion of course, students will be able to: 1. Understand the fundamentals of Automata theory and regular sets 2. Design regular language modelling in Finite Automata and non-deterministic finite automata 3. Understand context free and context sensitive languages and design automata for these languages 4. Understand undecidability and complexity of problems.					
<b>Detailed Syllabus:</b>					
Unit	Description				Duration
1	<b>Introduction:</b> Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages				04
2	<b>Regular languages and finite automata:</b> Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, Kleene's theorem, pumping lemma for regular languages, Myhill-Nerode theorem and its uses, minimization of finite automata.				12
3	<b>Context-free languages and pushdown automata:</b> Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic pushdown automata, closure properties of CFLs.				09

Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 1



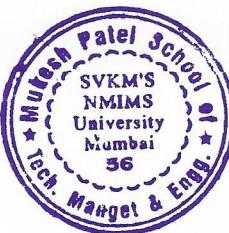
**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

4	Context-sensitive languages: Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.	04
5	Turing machines: The basic model for Turing machines (TM), Turing recognizable (recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.	07
6	Undecidability: Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice's theorem, undecidable problems about languages.	05
7	Basic Introduction to Complexity: Introductory ideas on Time complexity of deterministic and nondeterministic Turing machines, P and NP, NP-completeness, Cook's Theorem, other NP -Complete problems.	04
	<b>Total</b>	<b>45</b>
<b>Text Books:</b>		
1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman., "Introduction to Automata Theory, Languages, and Computation", 3rd Edition Pearson Education 2008.		
<b>Reference Books:</b>		
1. Harry R. Lewis and Christos H. Papadimitriou, "Elements of the Theory of Computation", Pearson Education, 2015 2. Dexter C. Kozen, Springer Verlag, "Automata and Computability", 3rd Printing 1999 3. Michael Sipser", "Introduction to the Theory of Computation", 3rd Edition, 2012 4. John Martin, "Introduction to Languages and the Theory of Computation", Tata Mac Graw Hill, 2007 5. M. R. Garey and D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP Completeness", Published by W.H. Freeman and Company, 1979		

  
Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III/A. Y 2020-21 -Page 2



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

**Any other information:**

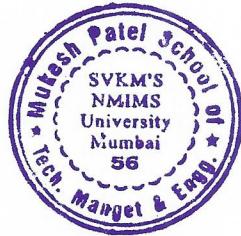
Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
<b>Total Marks :</b>	<b>50</b>

Signature  
(Prepared by Concerned Faculty/HOD)

Signature  
(Approved by Dean)



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

<b>Program:</b> B. Tech. Computer Science & Business Systems				<b>Semester:</b> III	
<b>Course/Module:</b> Computer Organization & Architecture				<b>Module Code:</b> BTCS03002	
<b>Teaching Scheme</b>				<b>Evaluation Scheme</b>	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks - 50)	Term End Examinations (TEE) (Marks- 100 in Question Paper)
3	2	0	4	Marks Scaled to 50	Marks Scaled to 50
<b>Pre-requisite:</b> Principles of Electronics Engineering (BTCS02004)					
<b>Objectives:</b> To provide knowledge of the basic principles of the organization, operation and performance of modern day computer systems and the underlying integrated circuit technologies that is used to construct computer components. To understand the use of parallel organization concepts in combining those components					
<b>Outcomes:</b> After successful completion of this course, students will be able to: <ol style="list-style-type: none"> <li>Understand the operation of functional blocks and buses</li> <li>Apply various operations of ALU techniques and memories</li> <li>Analyze Processor performance using various components of CPU</li> <li>Analyze various I/O operations and classifications</li> </ol>					

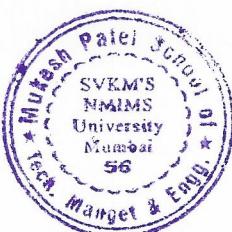
**Detailed Syllabus:**

Unit	Description	Duration
1	Revision of basics in Boolean logic and Combinational/Sequential Circuits.	02
2	Functional blocks of a computer: Overview of CPU, memory, input-output subsystems, control unit.	04
3	Data representation: Signed number representation, fixed and floating point representations, character representation.	03
4	Computer arithmetic: Integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication - shift-and-add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic, IEEE 754 format.	06



Signature  
 (Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 4



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

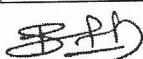
5	<b>Introduction to x86 architecture.</b>  Instruction set architecture of a CPU: Registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Outlining instruction sets of some common CPUs.	06
6	<b>Memory system design:</b> Semiconductor memory technologies, memory organization	04
7	<b>Memory organization:</b> Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.	07
8	<b>CPU control unit design:</b> Hardwired and micro-programmed design approaches, design of a simple hypothetical CPU.	05
9	<b>Peripheral devices and their characteristics:</b> Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB	05
10	<b>Pipelining:</b> Basic concepts of pipelining, throughput and speedup, pipeline hazards.  <b>Parallel Processors:</b> Introduction to parallel processors, Concurrent access to memory and cache coherency.	03
	<b>Total</b>	<b>45</b>

**Text Books:**

1. M. M. Mano, "Computer System Architecture", 3<sup>rd</sup> Edition., Prentice Hall of India, New Delhi, 1993.
2. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", 5<sup>th</sup> Edition, 2014
3. Carl Hamacher, "Computer Organization and Embedded Systems", 6<sup>th</sup> Edition, 2012

**Reference Books:**

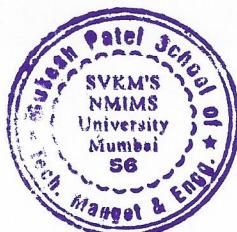
1. John P. Hayes, "Computer Architecture and Organization", 2<sup>nd</sup> Edition 2010
2. William Stallings, "Computer Organization and Architecture: Designing for Performance", 10<sup>th</sup> Edition Prentice Hall, 2016



Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III/A. Y 2020-21 -Page 5



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

3. Vincent P. Heuring and Harry F. Jordan, "Computer System Design and Architecture", 2<sup>nd</sup> Edition Prentice Hall, 2004

Any other information:

Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
<b>Total Marks :</b>	<b>50</b>

Practical Tool: Suitable simulators should be used.

Signature  
(Prepared by Concerned Faculty/HOD)

Signature  
(Approved by Dean)



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

<b>Program:</b> B. Tech. Computer Science & Business Systems <b>Course/Module:</b> Object Oriented Programming				<b>Semester:</b> III <b>Module Code:</b> BTCO03012	
<b>Teaching Scheme</b>				<b>Evaluation Scheme</b>	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks - 50)	Term End Examinations (TEE) (Marks- 100 in Question Paper)
3	2	0	4	Marks Scaled to 50	Marks Scaled to 50
Pre-requisite: Fundamentals of Computer Science (BTCS01003)					
<b>Objectives:</b> <ol style="list-style-type: none"> <li>1. The course will introduce standard tools and techniques for software development using object oriented approach.</li> <li>2. Enable students to solve complex problem using the knowledge of object oriented programming</li> </ol>					
<b>Outcomes:</b> After successful completion of this course, students will be able to <ol style="list-style-type: none"> <li>1. Develop programs using concept of Arrays, Pointers and structures.</li> <li>2. Develop object oriented programs using encapsulation, inheritance, polymorphism.</li> <li>3. Implement generic programming and file handling</li> <li>4. Design a UML diagram for a given problem statement.</li> </ol>					
<b>Detailed Syllabus:</b>					
Unit	Description				Duration
1	Procedural programming, An Overview of C: Types Operator and Expressions, Scope and Lifetime, Constants, Pointers, Arrays, and References, Control Flow, Functions and Program Structure, Namespaces, error handling, Input and Output (C-way), Library Functions ( <i>string, math, stdlib</i> ), Command line arguments, Pre-processor directive				05
2	Some difference between C and C++: Single line comments, Local variable declaration within function scope, function declaration, function overloading, stronger type checking, Reference variable, parameter passing – value vs reference, passing pointer by value or reference, Operator new and delete, the typecasting operator, Inline Functions in contrast to macro, default arguments				06

Signature  
(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III/A. Y 2020-21 -Page 7



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

3	The Fundamentals of Object Oriented Programming: Necessity for OOP, Data Hiding, Data Abstraction, Encapsulation, Procedural Abstraction, Class and Object.	05
4	More extensions to C in C++ to provide OOP Facilities: Scope of Class and Scope Resolution Operator, Member Function of a Class, private, protected and public Access Specifier, this Keyword, Constructors and Destructors, friend class, error handling (exception)	08
5	Essentials of Object Oriented Programming: Operator overloading, Inheritance - Single and Multiple, Class Hierarchy, Pointers to Objects, Assignment of an Object to another Object, Polymorphism through dynamic binding, Virtual Functions, Overloading, overriding and hiding, Error Handling	07
6	Generic Programming: Template concept, class template, function template, template specialization	03
7	Input and Output: Streams, Files, Library functions, formatted output	03
8	Object Oriented Design and Modelling: UML concept, Use case for requirement capturing, Class diagram, Activity diagram and Sequence Diagram for design, Corresponding C++ code from design	08
		45

**Text Books:**

1. Bjarne Stroustrup, "The C++ Programming Language", Addison Wesley, 4<sup>th</sup> Edition, 2013.
2. Debasish Jana, "C++ and Object-Oriented Programming Paradigm", PHI Learning Pvt. Ltd, 3<sup>rd</sup> Edition, 2014.

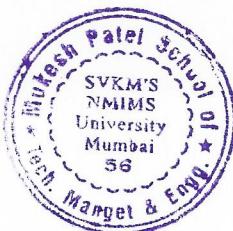
**Reference Books:**

1. Bjarne Stroustrup, "Programming - Principles and Practice Using C++", Addison Wesley, 2<sup>nd</sup> Edition, 2014.
2. Bjarne Stroustrup, "The Design and Evolution of C++", Addison Wesley, 4<sup>th</sup> Edition, 1994.

  
Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 8



SVKM's NMIMS  
Mukesh Patel School of Technology Management & Engineering

Any other information:  
Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

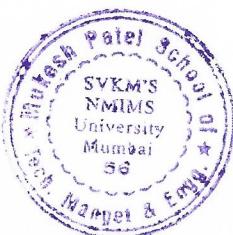
Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
Total Marks :	50



Signature  
(Prepared by Concerned Faculty/HOD)



Signature  
(Approved by Dean)



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

Program: B. Tech. Computer Science & Business Systems				Semester: III	
Course/Module: Computational Statistics				Module Code: BTCS03004	
Teaching Scheme				Evaluation Scheme	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks - 50)	Term End Examinations (TEE) (Marks- 100 in Question Paper)
3	2	0	4	Marks Scaled to 50	Marks Scaled to 50

Pre-requisite: Statistical Modelling BTCS01002

**Objectives:**

1. To develop sound knowledge and skills in theoretical, computational and application-oriented statistics.
2. To equip the students with intermediate to advanced level concepts and tools in statistics that help them tackle relevant problems within engineering domain.

**Outcomes:** After completion of the course, students would be able to:

1. Solve problems involving multivariate normal distribution, multiple and multivariate regression.
2. Demonstrate the ability to understand the concepts of Discriminant analysis, PCA, Factor analysis and Cluster analysis.
3. Analyze data samples using Python programming.

**Detailed Syllabus:**

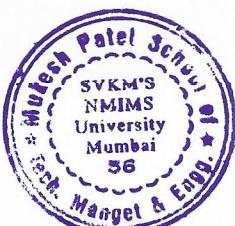
Unit	Description	Duration
1	<b>Multivariate Normal Distribution:</b> Multivariate Normal Distribution Functions, Conditional Distribution and its relation to regression model, Estimation of parameters.	7
2	<b>Multiple Linear Regression Model:</b> Standard multiple regression models with emphasis on detection of collinearity, outliers, non-normality and autocorrelation, Validation of model assumptions.	7
3	<b>Multivariate Regression:</b> Assumptions of Multivariate Regression Models, Parameter estimation, Multivariate Analysis of variance and covariance	7
4	<b>Discriminant Analysis:</b> Statistical background, linear discriminant function analysis, Estimating linear discriminant functions and their properties.	5



Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS / 2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 10



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

5	<b>Principal Component Analysis:</b> Principal components, Algorithm for conducting principal component analysis, deciding on how many principal components to retain, H-plot.	4
6	<b>Factor Analysis:</b> Factor analysis model, Extracting common factors, determining number of factors, Transformation of factor analysis solutions, Factor scores.	5
7	<b>Cluster Analysis:</b> Introduction, Types of clustering, Correlations and distances, clustering by partitioning methods, hierarchical clustering, overlapping clustering, K-Means Clustering-Profiling and Interpreting Clusters.	5
8	<b>Python Concepts, Data Structures, Classes:</b> Interpreter, Program Execution, Statements, Expressions, Numeric Types, Flow Controls, Functions, Sequences and Class Definition, Constructors, Text & Binary Files - Reading and Writing <b>Visualization in Python:</b> Matplotlib package, Plotting Graphs, Controlling Graph, Adding Text, More Graph Types, Getting and setting values, Patches (To be done in Lab session.) <b>Multivariate data analysis:</b> Multiple regression, multi variate regression, cluster analysis with various algorithms, factor analysis, PCA and linear discriminant analysis. (To be done in Lab session.)	5
<b>Total</b>		<b>45</b>

**Text Books:**

1. T.W. Anderson, "An Introduction to Multivariate Statistical Analysis", 3<sup>rd</sup> Edition, 2003.
2. J.D. Jobson, "Applied Multivariate Data Analysis", Vol I & II, Springer Publication, 1<sup>st</sup> Edition, 1999.
3. Mark Lutz, "Programming Python", O'Reilly, 4<sup>th</sup> Edition, 2010.
4. Tim Hall and J-P Stacey, "Python 3 for Absolute Beginners", Apress, 1<sup>st</sup> Edition, 2009.
5. Magnus Lie Hetland, "Beginning Python: From Novice to Professional", Edition, 2005.

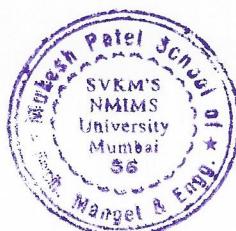
**Reference Books:**

1. D.A. Belsey, E. Kuh and R.E. Welsch, " Regression Diagnostics, Identifying Influential Data and Sources of Collinearity" John Wiley & Sons, 1989.
2. J. Neter, W. Wasserman and M.H. Kutner, " Applied Linear Regression Models", Mc Graw Hill Education, 4<sup>th</sup> Edition, 2004.
3. D.C. Montgomery & E. Peck, "Introduction to Linear Regression Analysis", John Wiley and Sons, 5<sup>th</sup> Edition, 2012
4. M.R. Anderberg, " Cluster Analysis for Applications", Springer NY, 1<sup>st</sup> Edition, 2000.
5. D.F. Morrison, " Multivariate Statistical Methods", Thomson/Brooks/Cole, 2005.

Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS / 2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 11



SVKM's NMIMS  
Mukesh Patel School of Technology Management & Engineering

6. Wes Mc Kinney," Python for Data Analysis", O'Reilly,2<sup>nd</sup> Edition,2017.

Any other information:

Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
<b>Total Marks :</b>	<b>50</b>

Signature  
(Prepared by Concerned Faculty/HOD)

Signature  
(Approved by Dean)



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

<b>Program:</b> B. Tech. Computer Science & Business Systems				<b>Semester:</b> III	
<b>Course/Module:</b> Software Engineering				<b>Module Code:</b> BTCS03005	
<b>Teaching Scheme</b>				<b>Evaluation Scheme</b>	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks - 50)	Term End Examinations (TEE) (Marks- _50 in Question Paper)
3	2	1	5	Marks Scaled to 50	Marks Scaled to 50

**Pre-requisite:** Fundamentals of Computer Science BTCS01003

**Objectives:** To impart the knowledge of Engineering approach to software development and project management. To impart knowledge of Software Building Techniques. Make students understand the Importance of Software quality and reliability. To make students understand the importance of Analysis, design and construction (both structured and object oriented). To impart the knowledge of Software Testing.

**Outcomes:** After successful completion of this course, students will be able to

1. Understand the concepts of software engineering.
2. Apply project management skills to manage the project.
3. Apply Design and analysis for the software to be built.
4. Implement and test the software project.

**Detailed Syllabus:**

Unit	Description	Duration
1	<b>Introduction:</b> Programming in the small vs. programming in the large; software project failures and importance of software quality and timely availability; engineering approach to software development; role of software engineering towards successful execution of large software projects; emergence of software engineering as a discipline.	5
2	<b>Software Project Management:</b> Basic concepts of life cycle models – different models and milestones; software project planning – identification of activities and resources; concepts of feasibility study; techniques for estimation of schedule and effort; software cost estimation models and concepts of software engineering economics; techniques of software project control and reporting; introduction to measurement of software size; introduction to the concepts of risk and its mitigation; configuration management.	8



Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III / A. Y 2020-21 -Page 13



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

3	<b>Software Quality and Reliability:</b> Internal and external qualities; process and product quality; principles to achieve software quality; introduction to different software quality models like McCall, Boehm, FURPS / FURPS+, Dromey, ISO - 9126; introduction to Capability Maturity Models (CMM and CMMI); introduction to software reliability, reliability models and estimation.	8
4	<b>Software Requirements Analysis, Design and Construction:</b> Introduction to Software Requirements Specifications (SRS) and requirement elicitation techniques; techniques for requirement modeling – decision tables, event tables, state transition tables, Petri nets; requirements documentation through use cases; introduction to UML, introduction to software metrics and metrics based control methods; measures of code and design quality.	8
5	<b>Object Oriented Analysis, Design and Construction:</b> Concepts -- the principles of abstraction, modularity, specification, encapsulation and information hiding; concepts of abstract data type; Class Responsibility Collaborator (CRC) model; quality of design; design measurements; concepts of design patterns; Refactoring; object oriented construction principles; object oriented metrics.	8
6	<b>Software Testing:</b> Introduction to faults and failures; basic testing concepts; concepts of verification and validation; black box and white box tests; white box test coverage – code coverage, condition coverage, branch coverage; basic concepts of black-box tests – equivalence classes, boundary value tests, usage of state tables; testing use cases; transaction based testing; testing for non-functional requirements – volume, performance and efficiency; concepts of inspection.	8
<b>Total</b>		<b>45</b>

**Text Books:**

1. Ian Sommerville, "Software Engineering", 9<sup>th</sup> edition, 2011.

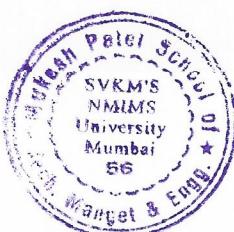
**Reference Books:**

1. *Fundamentals of Software Engineering*, Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino, 2<sup>nd</sup> edition, 2002
2. *Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices*, Michael Jackson, 2004.

Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III/ A. Y 2020-21 -Page 14



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

- |   |
|---|
| 3. <i>The Unified Development Process</i> , Ivar Jacobson, Grady Booch, James Rumbaugh, 2 <sup>nd</sup> edition, 2010                     |
| 4. <i>Design Patterns: Elements of Object-Oriented Reusable Software</i> , Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 2003 |
| 5. <i>Software Metrics: A Rigorous and Practical Approach</i> , Norman E Fenton, Shari Lawrence Pfleeger, 2015                            |
| 6. <i>Software Engineering: Theory and Practice</i> , Shari Lawrence Pfleeger and Joanne M. Atlee, 2013                                   |
| 7. <i>Object-Oriented Software Construction</i> , Bertrand Meyer, 2014  |
| 8. <i>Object Oriented Software Engineering: A Use Case Driven Approach</i> --Ivar Jacobson,2005   |
| 9. <i>Touch of Class: Learning to Program Well with Objects and Contracts</i> --Bertrand Meyer,2009                                       |
| 10. <i>UML Distilled: A Brief Guide to the Standard Object Modeling Language</i> --Martin Fowler,2018                                     |

**Any other information:**

Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

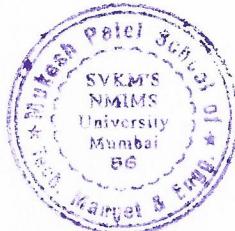
Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
<b>Total Marks :</b>	<b>50</b>

**Laboratory work:**

*Development of requirements specification, function oriented design using SA/SD, object-oriented design using UML, test case design, implementation using C++ and testing. Use of appropriate CASE tools and other tools such as configuration management tools, program analysis tools in the software life cycle.*

  
 Signature  
 (Prepared by Concerned Faculty/HOD)

  
 Signature  
 (Approved by Dean)



**SVKM's NMIMS**  
**Mukesh Patel School of Technology Management & Engineering**

<b>Program:</b> B Tech. Computer Science & Business Systems				<b>Semester:</b> III	
<b>Course/Module:</b> Indian Constitution				<b>Module Code:</b> BTCS03006	
<b>Teaching Scheme</b>				<b>Evaluation Scheme</b>	
Lecture (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Internal Continuous Assessment (ICA) (Marks 50)	Term End Examinations (TEE) (Marks- 100 in Question Paper)
2	0	0	0	Marks Scaled to 50	-

**Pre-requisite:**

**Objective:**

1. To understand the basic aspects of the constitution of India, the evolution, the directive principle & important provisions.
2. To understand the implications of important constitutional provision on Business and Professionals.

**Outcomes:** After completion of the course, students would be able to:

1. Describe basic aspects of constitution of India.
2. Understand Constitutional provision on Business and their Professionals.

**Detailed Syllabus:**

Unit	Description	Duration
1	The Constitution, its evolution and Preamble to the Constitution.	04
2	Fundamental rights and duties, exceptions with examples, individual responsibilities and duties, application to business.	10
3	Directive principles of State Policy, its emphasis and its impact as related to business.	04
4	Indian Judiciary and LokAdalats.	06
5	Emergency Provisions under Article 352 – 360.	04
6	Voting behaviour in India and present political scene. Responsibility of Business in relation to the Constitution.	02
	<b>Total</b>	<b>30</b>

**Text Books:**

1. Durga Das Basu, "Indian Constitution", 20<sup>th</sup> Edition, 2009.

**Reference Books:**

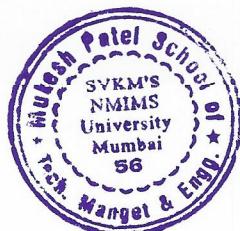
1. N. A. Palkhiwala, "We the People", 2009.
2. Justice Hidayatullah, "Indian Constitution", 2009.

**Any other information:**

Signature

(Prepared by Concerned Faculty/HOD)

B. Tech CSBS /2<sup>nd</sup> Year -Sem III/A. Y 2020-21 -Page 16



SVKM's NMIMS  
Mukesh Patel School of Technology Management & Engineering

Total Marks of Internal Continuous Assessment (ICA): 50 Marks

Distribution of ICA Marks:

Description of ICA	Marks
Class Test 1	10
Class Test 2	10
Term Work	30
<b>Total Marks :</b>	<b>50</b>



Signature  
(Prepared by Concerned Faculty/HOD)



Signature  
(Approved by Dean)

