Shri Acharyaratna Deshbhooshan Shikshan Prasarak Mandal's

Mahavir Mahavidyalaya, Kolhapur (Autonomous)

Affiliated to Shivaji University, Kolhapur

Accredited by NAAC with 'A' Grade

Programme with Multiple Entry and Multiple Exit Option As Per NEP-2020 (NEP-2.0)

Syllabus for

B.Sc. Computer Science (Entire) -I

Under the Faculty of Science

(To be introduced from Academic Year 2025– 26)

Subject to the revisions & modifications made from time to time)

B.Sc. Computer Science (Entire)

(Under Faculty of Science and Technology)

Program Outcomes (PO):

Upon successful completion of the B.Sc. Computer Science (Entire), the student should have met the following Outcomes:

PO1	Disciplinary Knowledge: Graduates will gain in-depth understanding in their specific major or discipline, mastering the foundational principles and theories, as well as advanced concepts. Execute strong theoretical and practical understanding developed from the specific programme in the area of work.
PO2	Problem-Solving Skills: Graduates will learn to use their knowledge to identify, analyze, and solve problems related to their field of study.
PO3	Analytical Skills: Graduates will gain the ability to collect, analyze, interpret, and apply data in a variety of contexts. They might also learn to use specialized software or equipment.
PO4	Research Skills and Scientific temper: Depending on the field, graduates might learn how to design and conduct experiments or studies, analyze results, and draw conclusions. They might also learn to review and understand academic literature.
PO5	Communication Skills: Many programs emphasize the ability to communicate effectively, both orally and in writing. Graduates may learn to present complex information clearly and succinctly, write detailed reports, and collaborate effectively with others.
PO6	Ethics and Professionalism: Graduates may learn about the ethical and professional standards in their field, and how to apply them in real-world situations.
P07	Integration: Integrate knowledge of Computer Science with associated subjects like mathematics, statistics, electronics etc. to build and explore problem solving concepts.

Program Specific Outcomes (PSO):

PSO1	Technical Expertise: Implement fundamental knowledge of core and programming computer subjects like C programming, operating system etc. For developing effective technical and computing solutions by incorporating creativity and logical reasoning.
PSO2	Successful Career: Deliver professional services and knowledge with updated new technologies like, Python, HTML, and PHP etc. in Computer science career.
PSO3	Interdisciplinary and Life Long Learning: Develop Mathematical and Electronical, Computation abilities. It also develops analytical, reasoning and logical abilities of students. Undergo higher studies, certifications and technology research as per market needs.
PSO4	Human Values and Ethics: Understand professional and ethical responsibilities in order to work at different positions in organizations and at a societal context.

1. **Introduction**

a) The name of the program shall be B.Sc. Computer Science (Entire).

b) After completion students will be able to apply standard software engineering practices and strategies in software project development using an open-source programming environment to deliver a quality product for business success.

c) Job Opportunities: The program addresses the job requirements in many domains such as web development, mobile development, Testing and one involving an assortment of hardware and software.

d) Many graduates begin their careers as junior programmers and, after some experience, are promoted as system analysts. Others seek a n entrepreneurial role in the Information Technology world as independent business owners, software authors, consultants, or suppliers of systems and equipment.

e) Career opportunities exist in such areas as software development and hardware integration, technical writing, training others on a computer, software design, software testing and technical support.

f) The present curricula focus on the learning aspect from three dimensions viz. Conceptual Learning, Skills Learning and Practical / Hands-on.

2. **Medium of Instruction:**

The medium of instruction will be English only.

3. **Admission Procedure**

To be eligible for admission to the B. Sc. Computer Science [Entire] a candidate must have passed

- HSC (10+2) from science stream

OR

- Three Year Diploma Course (after SSC i.e. 10^{th} Standard), of Board of Technical Education conducted by Government of Maharashtra or its equivalent

4. **Course Structure:**

Lectures and Practical should be conducted as per the scheme of lectures and practical's indicated in the course structure.

5. **Teaching and Practical Scheme**

a) Contact session for teaching 60 minutes each.

b) One Practical Batch should be of 20 students.

c) Practical evaluation should be conducted after the commencement of university examination.

6. **Assessment**

1. The final practical examination will be conducted by the university appointed examiners internal as well as external at the end of semester for each lab course and marks will be submitted to the universityby the panel.
2. The practical examination will be conducted semester wise in order to maintain the relevance of the respective theory course with laboratory course.
3. The final examinations shall be conducted at the end of the semester.
4. Nature of question paper: Nature of question paper is as follows for University end semester examination.

❖ **Theory Examination**:

Que. No.	Question		Marks
Q.1.	Multiple Choice Questions	(01 Mark each)	**06Marks**
Q.2.	Answer the question	(01 marks each)	**6 Marks**
Q.3	Attempt any TWO out of THREE	(03 marks each)	**6 Marks**
Q.4.	Long answer. Two out of Five	(6 marks each)	**12 Marks**
		Total Marks	**30 Marks**

- **Internal Evaluation examination of 20 marks should be in the form of assignments**.

❖ **Practical Examination**:

1. Practical Examination will be conducted at the end of Semester.
2. Each question paper carries **50 Marks**.
3. Duration of Practical Examination**: 3 Hrs**.
4. Nature of Question paper: There will be four questions of 20 marks each. Students will be attempted any two out of four questions. The distribution of practical's papers:

Each question carries	: 20 marks (20 x 02 = 40 Marks)
Certified Journal carries	: 5 Marks and
Viva voce carries	: 5 Marks
Total Marks	**: 50 Marks**

7. Standard of Passing:

1. Minimum 14 marks in each subject. There shall be separate passing for theory (semester end exam and Internal) and practical also.
2. Admission to B.Sc. Computer Science (Entire) Part II is allowed even if the student fails in all the subjects of part I
3. Admission to B.Sc. Computer Science (Entire) Part III is allowed only if student is passed on all the subjects of B.Sc. Computer Science (Entire) Part I

8. Board of Paper Setters /Examiners:

For each Semester end examination there will be a board of Paper setters and examiners for every course. While appointing paper setter /examiners, care should be taken to see that there is at least one person specialized in each unit of the course.

9. Credit system implementation:

As per the University norms

10. Clarification of Syllabus:

The syllabus committee should meet at least once in a year to study and clarify any difficulties from the Institutes.

11. Eligibility of Faculty:

MCA (from any faculty) or M.Sc. (Computer Science) with at least B+ or equivalent

12. Revision of Syllabus:

As the computer technology experience rapid rate of obsolescence of knowledge, revision of the syllabus should be considered every two/three years.

13. Fees Structure: As approved by the Shivaji University fee fixation committee.

14. Intake Capacity: 80

15. Award of Class:

Grading: Shivaji University has introduced a Seven-point grading system as follows:

B.Sc. Computer Science (Entire) Part I Semester I & II

Multiple Entry and Multiple Exit Option

(NEP-2020) 2.0

Syllabus to be implemented from Academic Year 2025-26

Sr. No.	Marks Range out of 50	Grade Point	CGPA	Letter grade
1.	43-50	10	9.50-10.00	O: Outstanding
2.	38-42	9	8.50-9.49	A+:Excellent
3.	33-37	8	7.50-8.49	A:Very Good
4.	28-32	7	6.50-7.49	B+:Good
5.	23-27	6	5.50-6.49	B: Average
6.	18-22	5	4.50-5.49	C:Satisfactory
7.	0-17	0	0.0-4.49	F:Fail
8.	Absent	0	------	-----

B.Sc. Computer Science (Entire) Part – I: Semester I & II

Multiple Entry and Multiple Exit Option (NEP-2020)

Syllabus to be implemented from Academic Year 2025-26

Title: **B.Sc. Computer Science (Entire)**

1. Year of implementation: Syllabus will be implemented from June 2024 onwards
2. Duration: B.Sc. Computer Science (Entire) Part I. The duration of course shall be one year (Twosemesters).
3. Pattern: Pattern of examination will be semester
4. Medium of Instruction: English
5. Structure Of Course:

Multiple Entry and Multiple Exit Option (NEP-2020)

B.Sc. Computer Science (Entire) Program Structure

B.Sc. Computer Science (Entire) Part - I (Level-4.5)

Semester	Subject Type	Course Code	Course Title
SEM – I	Course I:	Subject I DSC I:	C Programming
		Subject I DSC II:	Operating System
		Subject I Practical I:	C Programming Lab
	Course II:	Subject II DSC I:	Discrete Mathematics for Computer Science
		Subject II DSC II:	Algebra
		Subject II Practical I:	Mathematics Laboratory course–I
	Course III:	Subject III DSC I:	Fundamental Electronics
		Subject III DSC II:	Basic Digital Electronics
		Subject III Practical I:	Practical Based on Subject III DSC-I and Subject III DSC-II
	OE - I		Business Statistics Using MS Excel / Linux Practical - I
	IKS - I		Vedic Mathematics
SEM – II	Course I:	Subject I DSC III:	Advanced C Programming
		Subject I DSC IV:	Essentials of Software Engineering
		Subject I Practical II:	Advanced C Programming Lab
	Course II:	Subject II DSC III:	Graph theory
		Subject II DSC IV:	Group and Coding theory
		Subject II Practical II:	Mathematics Laboratory course–II
	Course III:	Subject III DSC III:	Sensors and Signal Conditioning
		Subject III DSC IV:	Advanced Digital Electronics
		Subject III Practical II:	Practical Based on Subject III DSC-III and Subject III DSC-IV
	OE - II		Business Statistics Using MS Excel / Linux Practical - II
	VEC - I		Democracy, Election and Constitution

SEMESTER-I (Duration - Six Month)										
Sr. No.	Course Code	Teaching Scheme			Examination Scheme					
		Theory and Practical			University Assessment (UA)			Internal Assessment (IA)		
		Lectures (Per week)	Hours (Per week)	Credit	Maximum Marks	Minimum Marks	Exam minutes	Maximum Marks	Minimum Marks	Exam. Hours
1	Subject I DSC I:	2	-	2	30	12	90	20	08	-
2	Subject I DSC II:	2	-	2	30	12	90	20	08	-
3	Subject I Practical I:	-	4	2	30	12	90	20	08	-
4	Subject II DSC I:	2	-	2	30	12	90	20	08	-
5	Subject II DSC II:	2	-	2	30	12	90	20	08	-
6	Subject II Practical I:	-	4	2	30	12	90	20	08	-
7	Subject III DSC I:	2	-	2	30	12	90	20	08	-
8	Subject III DSC II:	2	-	2	30	12	90	20	08	-
9	Subject III Practical I:	-	4	2	30	12	90	20	08	-
10	OE - I	-	4	2	30	12	--	20	08	90
11	IKS - I	2	-	2	30	12	90	20	08	-
	Total (A)			**22**	**330**			**220**	**330+220 = 550**	

SEMESTER-II (Duration- Six Month)										
Sr. No.	Course Code	Teaching Scheme			Examination Scheme					
		Theory and Practical			University Assessment (UA)			Internal Assessment (IA)		
		Lectures (Per week)	Hours (Per week)	Credit	Maximum Marks	Minimum Marks	Exam minutes	Maximum Marks	Minimum Marks	Exam. Hours
1	Subject I DSC III:	2	-	2	30	12	90	20	08	-
2	Subject I DSC IV:	2	-	2	30	12	90	20	08	-
3	Subject I Practical II:	-	4	2	30	12	90	20	08	-
4	Subject II DSC III:	2	-	2	30	12	90	20	08	-
5	Subject II DSC IV:	2	-	2	30	12	90	20	08	-
6	Subject II Practical II:	-	4	2	30	12	90	20	08	-
7	Subject III DSC III:	2	-	2	30	12	90	20	08	-
8	Subject III DSC IV:	2	-	2	30	12	90	20	08	-
9	Subject III Practical II:	-	4	2	30	12	90	20	08	-
10	OE - II	-	4	2	30	12	--	20	08	90
11	VEC - I	2	-	2	30	12	90	20	08	-
	Total (B)			**22**	**330**			**220**	**330 + 220 = 550**	
	Total (A+B)		**22+22 = 44**		**660**			**440**	**660 + 440 = 1100**	

• Student contact hours per week: **30** Hours (Min.)	• Total Marks for B.Sc. Computer Science (Entire)-I: **1100**
• Theory and Practical Lectures: **60** Minutes Each	• Total Credits for B.Sc. Computer Science (Entire)-I (Semester I & II): **44**
• **Requirement for Entry at Level 4.5: Completed all requirements of the 10+2**	
• DSC: Department Specific Core • OE: Open Elective	• IKS: Indian Knowledge System • VEC: Value Education Course
• Practical Examination is Semester wise before Theory Examination.	• Separate passing is mandatory for Theory, Internal and Practical Examination
• **Exit Option at Level 4.5:** Students can exit after Level 4.5 with under **Certificate Course in Computer Programming** if he/she completes the courses equivalent to minimum of **44** credits and an **additional 4 credits** core NSQF course / Internship.	

B.Sc. Computer Science (Entire) - Part I

DSC: Computer Science

Total Work–Load

Paper No.	Title of the Paper	Theory Marks	Internal Marks	Lectures / week (60 min.)
Semester I				
Subject I DSC I:	C Programming	30	20	2
Subject I DSC II:	Operating System	30	20	2
Subject II DSC I:	Algebra	30	20	2
Subject II DSC II:	Mathematics Laboratory course–I	30	20	2
Subject III DSC I:	Fundamental Electronics	30	20	2
Subject III DSC II:	Basic Digital Electronics	30	20	2
DSC Practical – I				
Paper No.	**Title of the Paper**	**Total Marks**	**Internal Marks**	**Lectures per week/ Batch**
Subject I Practical I:	C Programming Lab	40	10	4
Subject II Practical I:	Mathematics Laboratory course–I	40	10	4
Subject III Practical I:	Practical Based on Subject III DSC-I and Subject III DSC-II	40	10	4
OE - I	Business Statistics Using MS Excel / Linux Practical - I	40	10	2
IKS - I	Vedic Mathematics	40	10	2
Semester II				
Paper No.	**Title of the Paper**	**Total Marks**	**Internal Marks**	**Lectures / week (60 min.)**
Subject I DSC III:	Advanced C Programming	30	20	2
Subject I DSC IV:	Essentials of Software Engineering	30	20	2
Subject II DSC III:	Graph theory	30	20	2
Subject II DSC IV:	Group and Coding theory	30	20	2
Subject III DSC III:	Sensors and Signal Conditioning	30	20	2
Subject III DSC IV:	Advanced Digital Electronics	30	20	2
DSC Practical – II				
Paper No.	**Title of the Paper**	**Total Marks**	**Internal Marks**	**Lectures perweek / Batch**
Subject I Practical II:	Advanced C Programming Lab	40	10	4
Subject II Practical II:	Mathematics Laboratory course–II	40	10	4
Subject III Practical II:	Practical Based on Subject III DSC-III and Subject III DSC-IV	40	10	4
OE – II	Business Statistics Using MS Excel / Linux Practical - II	40	10	2
VEC - I	Democracy, Election and Constitution	40	10	2

B. Sc. Part- I Computer Science (Entire) (Semester I)

Paper No.	Title of the Paper	Theory Marks	Internal Marks	Total Marks
Semester I				
Paper No.	**Title of the Paper**	**Theory Marks**	**Internal Marks**	**Total Marks**
Subject I DSC I:	C Programming	30	20	50
Subject I DSC II:	Operating System	30	20	50
Subject II DSC I:	Algebra	30	20	50
Subject II DSC II:	Mathematics Laboratory course–I	30	20	50
Subject III DSC I:	Fundamental Electronics	30	20	50
Subject III DSC II:	Basic Digital Electronics	30	20	50
DSC Practical – I				
Paper No.	**Title of the Paper**	**Theory Marks**	**Internal Marks**	**Total Marks**
Subject I Practical I:	C Programming Lab	40	10	50
Subject II Practical I:	Mathematics Laboratory course–I	40	10	50
Subject III Practical I:	Practical Based on Subject III DSC-I and Subject III DSC-II	40	10	50
OE / IKS				
OE - II	Business Statistics Using MS Excel / Linux Practical - I	30	20	50
IKS - I	Vedic Mathematics	30	20	50

B. Sc. Part- I Computer Science (Entire) (Semester I)

Course Code: **Subject I DSC I:** Course Title**: C Programming**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand the concept of design tools (Algorithm and Flowchart) to given solution to the problem.

CO2 Use basics of C language syntax as identifiers, keywords, variables, data types and operators

CO3 Apply the concept of branching, looping, decision-making statements and Array for given problem.

CO4 Break a large problem into smaller part, writing each part as a function and develop a C Program.

Unit	Contents	Hours Allotted
1	**Introduction to C language fundamentals** • Problem solving process: Problem Analysis, Problem Design: Algorithm and Flowchart, Coding, Debugging and Testing • structure of C program, Hello World C Program, Compilation and Execution of C program (using gcc compiler) • Formatted functions with format specifiers: printf() and scanf() • Identifiers, keywords, Variables, Data types, type casting in C • Operators and Expressions, Types of errors in C	15
2	**Control Structures and Array** • Branching statements: If statement, If-else statement, If else-if ladder, Nested if, switch statement, ternary operator • Looping statements: for loop, while loop, do while loop, nested loop, infinite loop • Jumping statements: break, continue and goto statement • Array: Definition and Advantages of an Array, Array Declaration and Initialization, memory representation of an Array, Accessing array elements, 1-D Array, Multi-Dimensional. • Functions in C, types of functions: library function and user defined function, function declaration, function definition, and calling a function, simple program of addition of two numbers using user defined function.	15

Reference Books

1. The C Programming Language - By Brian W Kernighan and Dennis Ritchie
2. C programming in an open-source paradigm: By R. K. Kamat, K. S. Oza, S.R.Patil
3. The GNU C Programming Tutorial -By Mark Burgess
4. Let us C- By Yashwant Kanetkar

Course Code: **Subject I DSC II:** Course Title**: Operating System**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand basic concepts of operating system, services and their structures.

CO2 Illustrate the concept of process and process life cycle and acquire the knowledge of CPU and I/O concepts.

CO3 Implement the issues and challenges of memory management and file management concept

CO4 Understand the concept of resource allocation and concept of deadlock with its prevention, avoidance, detection and recovery.

Unit	Contents	Hours Allotted
1	**Introduction and Operating-Systems Structures**: Definition of Operating system, Operating System's role, Operating-System Operations, Functions of Operating System, Computing Environments, Operating-System Structures: Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, Operating-System Structure Processes: Process Concept, Process Scheduling, Operations on Processes Interprocess Communication. Threads: Overview, Multicore Programming, Multithreading Models. Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock	15
2	**Main Memory**: Background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing Mass-Storage Structure: Overview, Disk Structure, Disk Scheduling, Disk Management File-System Interface: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing File-System Implementation: File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management	15

Reference Books

1. Operating System Concepts – Silberschatz, Galvin and Gagne
2. Operating System By Achyutya Godbole

Course Code: **Subject I Practical I**: Course Title: **C Programming Lab**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand basic structure if C Programming, declaration and usage of variables, use of data type and operators.

CO2 Implement control structures to develop a C program.

CO3 Apply and write C Program to implement one dimensional array.

CO4 Define a user defined function to give solution to given problem.

List of Laboratory Assignments

1. Program based on input (printf()) and output(scanf()) functions.
2. Program based on operators and expressions
3. Program based on branching statements
4. Program based on switch statement
5. Program based on for loop statements
6. Program based on while statements
7. Program based on do while loop statements
8. Program based on break and continue statement
9. Program based on Array: Find maximum number between given array.
10. Program based on Array: display array in ascending order.
11. Program based on function: user defined function to add two integer numbers.
12. Program based on function: user defined function to find maximum number between two numbers.
13. Program based on function: user defined function to display square of a given number.

Course Code: **Subject II DSC I:** Course Title**: Discrete Mathematics for Computer Science**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Apply basic counting principles and combinatorial arguments.

CO2 Solve linear recurrence relations with constant coefficient.

CO3 Analyze the logical structure of statements symbolically, including the proper use of logical connectives.

CO4 Construct truth tables, prove or disprove a hypothesis and evaluate the truth of a statement using the principles of logic.

Unit	Contents	Hours Allotted
1	**Unit 1: Counting Principles** Functions: Definition, Types of mapping, Injective, Surjective & Bijective functions, Inverse function, Composition of functions. Counting: Addition & Multiplication principle, Permutation and Combination. Cardinality of finite set. Cardinality of union of sets (Addition principle). Principle of Inclusion and Exclusion. Examples. Combinatorial Arguments. Pigeonhole Principle (Statement only). Examples.	15
2	**Unit 2: Recurrence Relation & Logic** **Recurrence Relation** Introduction. Linear Recurrence relation with constant coefficient. Homogeneous solutions and Examples. Particular and Total Solution, Examples. **Logic** Propositions and Logical connectives: Definition, Types of Propositions, Truth values and Truth Tables, Tautology and Contradiction, Logical equivalence. Rules of inferences. Valid arguments and proofs. Methods of Proofs: Direct and indirect Examples.	15

Reference Books

1. Discrete mathematics by S. R. Patil and others, NIRALI Prakashan.
2. Discrete mathematics by Bhopatkar, Nimbkar, Joglekar, VISION Publication.
3. Discrete Mathematical Structure for Computer Science by Alan Doer and K. Levasicur.
4. Discrete Mathematics by Olympia Nicodemi.
5. Elements of Discrete Mathematics by C. L. Liu.
6. Discrete and Combinatorial Mathematics by R. M. Grassl.
7. Discrete Mathematics by Kenneth Rosen, Tata McGraw Hill.
8. Discrete mathematics by Naik and Patil, PHADAKE Prakashan.

Course Code: **Subject II DSC II:** Course Title**: Algebra**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Apply fundamental concepts in Number theory to solve problems on congruence.

CO2 Solve problems based on Fermat's theorem and residue classes.

CO3 Use fundamental concepts in Mathematics like sets, relations and functions.

CO4 learn basic concepts like poset, lattice, Boolean algebra and apply them to find CNF and DNF.

Unit	Contents	Hours Allotted
1	**Unit 1: Divisibility of integers** Introduction Divisibility: Division algorithm (Statement only). Greatest Common Divisor (GCD), Least Common Multiple (LCM), examples. Euclidean algorithm, examples. Prime numbers, Euclides Lemma, Fundamental theorem of Arithmetic (without proof), examples. Congruence relation and its properties Fermat's Theorem (Statement only), examples. Residue Classes: Definition, addition modulo n, multiplication modulo n, Examples.	15
2	**Unit 2: Relations & Boolean Algebra** **Relations** Ordered pairs, Cartesian product. Relations, Types of relations, Equivalence relation, Partial orderingrelation, Examples. Digraphs of relations, matrix representation and composition of Relations,Examples. Transitive closure, Warshall's algorithm, Examples. Equivalence class, Partition of a set. **Boolean algebra** Hasse diagram. Lattice: Definition, principle of duality. Basic properties of algebraic systems defined by Lattices. Distributive and complemented lattices. Boolean lattices and Boolean algebras. Boolean expressions and Boolean functions. Disjunctive and conjunctive normal forms and examples.	15

Reference Books

1. Algebra by S. R. Patil and Others Nirali Prakashan.
2. Algebra by Bhopatkar, Nimbkar, Joglekar, VISION Publication.

3. Algebra by Naik and Patil, PHADAKE Prakashan.
4. A Foundation Course in Mathematics, Ajit Kumar, S. Kumeresan and Bhaba Kumar Sarma, Narosa Publication House.
5. Elementary Number Theory, Seventh edition: David M. Burton, McGraw-Hill.
6. Lattices & Boolean Algebras: First Concepts by V. K. Khanna, Vikas Publishing House, Second Edition, 2008

Course Code: **Subject II Practical I**: Course Title: **Mathematics Laboratory course–I**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

Pr. No	**Title of the Practical**	**No. of Practical**
1.	Combinatorial arguments	1
2.	Recurrence relation	1
3.	Proofs of valid arguments using truth table	1
4.	Proofs of valid arguments using laws of inferences	1
5.	Euclidean algorithm	1
6.	Examples using Fermat's theorem	1
7.	Warshall's algorithm	1
8.	Disjunctive and Conjunctive normal forms (DNF & CNF)	1
9.	C – Programs: finding g. c. d and l. c. m., determination of primes	1
10.	C – program for Euclidean algorithm	1
11.	C – program for Warshall's algorithm	1
12.	C – program to determine the value of φ (*n*) (Euler φ function).	1

Batch: One batch of 20 students.

Course Code: **Subject III DSC I:** Course Title**: Fundamental Electronics**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1: Understand the concept of electronics components.

CO2: Understand the transistor Applications.

CO3: To study and understand the amplifier and oscillator concept.

CO4: To study the concept of operational amplifier and Integrated circuit.

Unit	Contents	Hours Allotted
1	**A) Linear & Non-linear Components in Computer:** **Resistors:** Classification of resistors, Symbols, color code method and its applications, **Capacitors**: Classification of capacitors, Symbols, electrolyte capacitor, applications of capacitor, **Inductors**: types of inductors, Symbols, its applications, **Diodes:** Types of diodes, Symbols, Forward bias & reverse biasing of a diode, Zener diode, LED diode, Applications of diodes, **B) Bipolar Junction Transistor:** Types of Transistors, Symbols, Construction details & working of NPN & PNP transistors, Operating modes of transistor, Biasing of a Transistor: Voltage divider bias, Emitter bias. Applications: Transistor as an Amplifier, Transistor as an Electronic Switch, Single stage amplifier & Need of Multistage amplifier, Coupling Scheme: Direct, RC, LC coupling in detail (only circuits using transistors & frequency response)	15
2	**A) Operational Amplifier:** Concept of Differential amplifier, Definition of Operational Amplifier, Internal block diagram of Op-Amp **IC-741**, Symbol & Pin diagram of IC-741**,** Ideal & Practical characteristics/parameters of Op-Amp IC-741, Configurations of Op-Amp: Open-Loop & Closed Loop, **B) Applications of Op-Amp:** Inverting mode amplifier, Virtual ground, Sign changer(Inverter), Non-inverting mode amplifier, Voltage follower(Unity gain buffer), Op-Amp Adder, Op-Amp-Subtractor, Op-Amp Comparators, Zero crossing detector, Timer IC-555: Pin diagram & internal block diagram of IC-555, Applications of IC-555: Astable multivibrator (duty cycle & frequency), Monostable multivibrator (pulse width calculation),	15

Reference Books

1. Principles of Electronics: by V. K. Meheta, S. Chand & Company Ltd.
2. Basic Electronics and Linear Circuits : by N. N. Bhargava, D. C. Kulshreshtha, S. C. Gupta,
3. Electronic Devices and circuits: by Robert Boylstead, Tata Mc-Graw Hill.
4. Linear Integrated Circuits: by Ramakant Gaikwad,
5. Principles of Electronics: by A.P.Malvino, Tata Mc-Graw Hill Publication,

Course Code: **Subject III DSC II:** Course Title**: Basic Digital Electronics**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1: Understand the concept of Number Systems,
CO2: Understand different Computer Codes,
CO3: Understand different Logic Gates & Boolean Algebra,
CO4: Understand various Combinational Logic circuits,

Unit	Contents	Hours Allotted
1	**A) Number Systems :** Introduction and definition, Classification (Weighted & Non-Weighted), Weighted Number System s: Binary Number System, Decimal Number System, Octal Number System, Hexadecimal Number System, Conversion of Numbers from one system to another system., Binary Arithmetic, 1's & 2's complement of binary numbers. Subtraction by 1's complement & 2's Complement. **B) Computer Codes & code conversion**: Introduction and definition, BCD code : (4-bit packed BCD, unpacked BCD), EBCDIC code , ASCII code: (ASCII – 7, ASCII-8), **Code Conversion**: Gray Code to Binary code, Binary to Gray code, Binary to Excess-3 code, Excess-3 code to Binary code, Concept of Parity bit, Even parity, Odd parity, Signed number and Unsigned number representation	15
2	**A) Logic Gates:** Definition, AND, OR, NOT, NOR, NAND, EX-OR (Symbol, Boolean Expression and Truth Table), Boolean algebra and Identities, De Morgan's theorems. Universal logic Gates (NAND and NOR), Boolean Equations, SOP expression & POS expression, Minterms & Maxterms, Introduction to K – map techniques with examples, **B) Combinational Logic Circuits:** Introduction, Half adder, Full Adder, Half Subtractor, Full Subtractor, Parallel adder, Universal Adder & Subtractor, Encoder (decimal to BCD), Priority Encoder, Decoder (BCD-Decimal), Multiplexer & De-multiplexer,	15

Reference Books

1. Digital Principles and Applications: by Malvino Leach, Tata McGraw Hill.
2. Fundamentals of Digital Electronics: by Anand Kumar, PHI Publication.
3. Digital Principles: by T. L. Floyd.
4. Digital Electronics: by R. P Jain.
5. Digital Electronics & Logic Design: by N. G. Palan,

Course Code: **Subject III Practical I:**

Course Title**: Practical Based on Subject III DSC-I and Subject III DSC-II**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

List of Laboratory Assignments

Sr. No.	**Name of the Practicals**
1	Study of various Electronic components, equipments & measuring devices.
2	Study of measurement of Amplitude, Frequency & Phase of waveforms by using CRO.
3	Study of PN junction diode (Forward biasing & Reverse Biasing).
4	Study of Transistor working as Electronic switch (Use LED & Relay in the circuit)
5	Study of Inverting mode Amplifier by using Op-Amp IC-741.
6	Study of Non-inverting mode amplifier by using Op-Amp IC-741.
7	Study of Op-Amp Adder by using IC741.
8	Study of Op-Amp Subtractor by using IC741.
9	Study of Astable Multivibrator by using IC555.
10	Study of Mono-stable Multivibrator by using IC555.
11	Study of Basic Logic gates.
12	Study of Universal Logic gate (NAND gate).
13	Study of Universal Logic gate(NOR gate).
14	Study of DeMorgan's Theorems.
15	Study of Half Adder.
16	Study of Full Adder.
17	Study of Half Subtractor,
18	Study of Full Subtractor.
19	Study of Multiplexer (4:1 or 8:1).
20	Study of DeMultiplexer (1:4 or 1:8).

Course Code**: OE-I:** Course Title: **Business Statistics Using MS Excel / Linux Practical - I**	Syllabus will be considered as per Shivaji University, Kolhapur

Course Code: **IKS - I**: Course Title: **Vedic Mathematics**

Total Contact Hours**: 30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes:

After completion of this course students will be able;

CO1: To perform simple arithmetic calculations with speed and accuracy

CO2: To generate tables of any number

CO3: To perform products of large numbers quickly

Unit	Contents	Hours Allotted
1	Introduction to Vedas, History of Vedas History and Evolution of Vedic Mathematics Introduction of Basic Vedic Mathematics Techniques in Multiplication (Special Case, Series of 9, Series of 1 etc.), Tables etc., Various techniques to carry out basic operations covering Addition, Subtraction, Multiplication, Division, Complements and Bases, Vinculum number. Comparison of Standard Methods with Vedic Methods.	15
2	General multiplication (Vertically Cross- wise), Multiplications by numbers near base. Verifying answers by use of digital roots, Divisibility tests, Division of numbers near base, Comparison of fractions. Different methods of Squares (General method, Base method, Duplex method etc.) Cubes, Cube roots, Square Roots, General division. Quadratic Equations, Simultaneous Equations, Use of various Vedic Techniques for answering numerical aptitude questions from Competitive Examinations.	15

Reference Books:

1. Bhatiya Dhaval, Vedic Mathematics Made Easy, Jaico Publishing House.
2. Thakur Rajesh Kumar, Vedic Mathematics for students taking Competitive Examinations. Unicorn Books 2015 or Later Edition.
3. Gupta Atul, Power of Vedic Mathematics with Trigonometry ,JaicoBooks
4. V. G. Unkalkar, Magical World of Mathematics (Vedic Mathematics), Vandana Publishers, Bangalore.
5. Bhatiya Dhaval, Vedic Mathematics Made Easy, Jaico Publishing House.
6. Thakur Rajesh Kumar, Vedic Mathematics for students taking Competitive Examinations. Unicorn Books 2015

Mahaveer Mahavidalaya, Kolhapur Faculty of Commerce and Management

Open Elective Course for Part I Semester – I of the UG Programmes of the Faculties Other Than Faculty of Commerce and Management

Course Name: Business Statistics using MS Excel/Linux
Practical – IPractical: 60 Hrs. Marks: 50
(Credits: 02)

Course Outcomes:

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

i. perform the visual analysis of data by means of simple diagrams and graphs, also to locate outliers using Excel functions
ii. get basic knowledge of descriptive statistics for data analysis
iii. get the basic knowledge of concepts of spread of data and exhibit variation in data by computing measures of dispersion.
iv. get the knowledge of type and shape of frequency distribution using skewness and kurtosis measures

List of Practical's:

1. Construction of Frequency Distribution and Graphical representation of data: Histogram, Frequency Polygon, Frequency Curve and Ogive Curves
2. Diagrammatical representation of data: Simple and Multiple Bar Diagram, Pie Chart, Stem and Leaf Plot
3. Computation of Measures of Central Tendency: Mean, Mode, Median, quartiles for ungrouped data
4. Computation of Measures of Central Tendency: Mean, Mode, Median, quartiles for grouped Data
5. Computation of Measures of Dispersion: Range, Quartile Deviation, Standard Deviation, Variance and their respective relative measures along with Coefficient of Variation (C.V.) for ungrouped data
6. Computation of Measures of Dispersion: Range, Quartile Deviation, Standard Deviation and their respective relative measures along with Coefficient of Variation (C.V.) for grouped data
7. Computation of Moments: first four raw and central moments for ungrouped data
8. Computation of Moments: first four raw and central moments for grouped data
9. Computation of Measures of Skewness and Kurtosis based on moments
10. Case study of at least 3 out of above practical's using primary data obtained by survey

Note:

- All practicals should conducted on Computer using MS Excel/Libre Office Calc

(Linux) software.

- Computer printout of each practical with output to be attached to the journal.
- Student must complete the entire practical to the satisfaction of the teacher concerned.
- Student must produce the laboratory journal along with the completion certificate signed by the Head of the department, at the time of practical examination.

Reference Books:

1. Agarwal B. L. (2019) *Basic Statistics*, New Age International (P) Limited.
2. Gupta S. C. (2019) *Fundamentals of Statistics*, Himalaya Publishing House Pvt. Ltd.
3. *A First Course in Probability* by Sheldon Ross (2022), Pearson pub.
4. *Statistical Methods* (An introductory text by J. Medhi), New Age International (P)Limited.
5. *Business Statistics: A First Course* by David Levine, Katherene szabat, Pearson Pub.
6. Sharma V. K. (2012) *Elements of Statistics*, Gullybaba Publishing House Pvt. Ltd.

Practical Examination:

1. Practical Examination will be conducted at the end of each Semester.
2. Each practical paper carries 50 Marks.
3. Duration of Practical Examination: 4 Hrs.

Nature of Question Paper:

i. There will be four questions of 18 marks each.
ii. In each question there are two sub questions (a) and (b) each carrying 09 marks
iii. Students have to attempt any two out of four questions.
iv. The distribution of practical paper's marks:

- Two questions each of 18 marks (Total 18 x 02 = 36 Marks)
- Certified Journal: 05 Marks,
- Viva voce: 04 Marks
- Case study: 05 marks
- Total Marks: 50

Course code : IKS I
Title of course : Vedic Mathematics
Theory 30
Marks 50
Credit 02

Course Outcomes:

1. By successfully completing this course, the learner will be able to:
2. Perform simple arithmetic calculations with speed and accuracy
3. Will be able to generate tables of any number

UNIT I (15 HOURS)

Introduction to Vedas, History of Vedas, History and Evolution of Vedic Mathematics Introduction of Basic Vedic Mathematics, Techniques in Multiplication, Special Case, Tables etc. Various techniques to carry out basic operations covering Addition, Subtraction, Multiplication, Division, Complements and Bases, Vinculum number. Comparison of Standard Methods with Vedic Methods

UNIT II (15 HOURS)

General multiplication (Vertically Cross- wise), Multiplications by numbers nearbase. Verifying answers by use of digital roots, Divisibility tests, Division of numbers near base, Comparison of fractions.Different methods of Squares (General method, Basemethod, Duplex method etc.)Cubes, Cube roots, Square Roots, General division.Quadratic Equations, Simultaneous Equations, Use of various Vedic Techniques for answering numerical aptitude questions from Competitive Examinations

Reference Books:

1. Bhatiya Dhaval, Vedic Mathematics Made Easy, Jaico Publishing House
2. Thakur Rajesh Kumar, Vedic Mathematics for students taking Competitive Examinations

B. Sc. Part- I Computer Science (Entire) (Semester II)

Paper No.	Title of the Paper	Theory Marks	Internal Marks	Total Marks
Semester II				
Paper No.	**Title of the Paper**	**Theory Marks**	**Internal Marks**	**Total Marks**
Subject I DSC III:	Advanced C Programming	30	20	50
Subject I DSC IV:	Essentials of Software Engineering	30	20	50
Subject II DSC III:	Graph theory	30	20	50
Subject II DSC IV:	Group and Coding theory	30	20	50
Subject III DSC III:	Sensors and Signal Conditioning	30	20	50
Subject III DSC IV:	Advanced Digital Electronics	30	20	50
DSC Practical – II				
Paper No.	**Title of the Paper**	**Theory Marks**	**Internal Marks**	**Total Marks**
Subject I Practical II:	Advanced C Programming Lab	40	10	50
Subject II Practical II:	Mathematics Laboratory course–II	40	10	50
Subject III Practical II:	Practical Based on Subject III DSC-III and Subject III DSC-IV	40	10	50
OE / VEC				
OE - II	Business Statistics Using MS Excel / Linux Practical - II	40	10	50
VEC - I	Democracy, Election and Constitution	40	10	50

B. Sc. Part- I Computer Science (Entire) (Semester II)

Course Code**: Subject I DSC III** Course Title**: Advanced C Programming**

Total Contact Hours: 30 hrs. (30 lectures of 60 min)

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Apply code reusability with functions and pointer, Implement string in C programs.
CO2 Understand how to allocate memory at runtime using different memory allocation functions.
CO3 Understand the need of structure and implement the structure with real life examples.
CO4 Understand the basics of file handling mechanism and uses of preprocessors.

Unit	Contents	Hours Allotted
1	**Pointers and Functions** • Declaring pointers, use of pointers, pointer to pointer. • Pointer arithmetic, array of pointers, passing pointers to functions. • Passing parameters to functions: call by value and call by reference. • Recursion in C, local and global variables. • Strings in C, string functions: strlen(),strcpy(),strcat(),strcmp(), strlwr(), strupr(),strrev() • Storage classes. • Dynamic memory allocation: malloc(),calloc(), realloc(),free().	15
2	**Structure, Union, and File Handling** • Introduction to Structure: declaring a structure, accessing members of astructure, array of structure. • Union: declaring a union, accessing members of a union. • Difference between structure and union. • File handling in C, creating a new file, opening an existing file (fopen(), fclose()), file opening modes(r, w, a, r+, w+, a+, rb, wb, ab, rb+, wb+, ab+), reading from the file, writing to the file, and appending the file using differentfile handling functions. • fprintf(), fscanf(), fputs(), fgets(), fputc() and fgetc(), fseek(),ftell() and rewind() functions in file handling. • Preprocessors in C: #include, #define, #undef, #ifdef, #ifndef, #if, #else, #elif, #endif, #error	15

Reference Books

1. The C Programming Language - By Brian W Kernighan and Dennis Ritchie
2. C programming in an open-source paradigm - By R. K. Kamat, K. S. Oza, S.R.Patil
3. The GNU C Programming Tutorial -By Mark Burgess
4. Let us C- By Yashwant Kanetkar

Course Code: **Subject I DSC IV:** Course Title**: Essentials of Software Engineering**

Total Contact Hours: 30 hrs. (30 lectures of 60 min)

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand the problem domain to choose process models correctly.

CO2 Choose software projects using appropriate design notations

CO3 Measure the product and process performance using various metrics.

CO4 Evaluate the system with various testing techniques and strategies

Unit	Contents	Hours Allotted
1	**Introduction to Software Engineering:** Introduction to Software, Definition and need for Software Engineering, Characteristics of good quality software, Software Development Life Cycle. Software Process Models: Linear Sequential Model, Prototyping Model, RAD Model, Incremental Model, Incremental Model, Spiral Model Software Metrics: Definition, Types of metrics: product metrics, process metrics, its advantages. **Software Project Planning** Software Project Planning Size Estimation, Cost Estimation and Time Estimation, Project scheduling and Tracking Software Design Process, Design Principles, SRS: introduction, characteristics of SRS.	15
2	**Software Design & Software Testing** **Software Design** Definition of Software Design, Software Design Process, Design Principles, DFD, Data Dictionary & ERD, Cohesion and coupling, Software Quality Assurance. **Software Testing** Software Testing Fundamentals White Box Testing, Black Box Testing Software testing strategies, Verification and Validation, System Testing, Unit testing, Integration testing and Debugging Implementation types, Software Maintenance, Maintenance Tasks	15

Text Book / Reference Books

1. Roger S Pressman, Bruce R Maxim, "Software Engineering: A Practitioner's Approach", Kindle Edition.
2. Ian Sommerville," Software engineering", Addison Wesley Longman, 2014.

Course Code: **Subject I Practical II**: Course Title: **Advanced C Programming Lab**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand how to reuse code using functions and pointers.
CO2 Implement memory allocation functions to allocate memory at run time.
CO3 Define a structure to declare the data members of different data types according to needs.
CO4 Handle different file handling functions and preprocessors.

List of Practicals:

1. Program based on declaration and use of a pointer.
2. Program based on the pointer of a pointer.
3. Program based on pointer arithmetic.
4. Program based on a call by value and call by reference.
5. Program based on recursive function.
6. Program based on storage classes.
7. Program based on string functions.
8. Program based on dynamic memory allocation.
9. Program based on structure.
10. Program based on file handling: creating a new file, writing(fputc()), and reading(fgetc()) the content of a file.
11. Program based on file handling: formatted printing to a file and formatted scanning from a file.
12. Program based on file handling: Binary file operations using fread() and fwrite()
13. Program based on file handling: Random Access File using ftell(), fseek(), and rewind()
14. Program based on file handling: copy the content of one file to another.
15. Program based on preprocessor.

Course Code: **Subject II DSC III:** Course Title**: Graph theory**

Total Contact Hours: 30 hrs. (30 lectures of 60 min)

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Achieve command of the fundamental definitions and concepts of graph theory.
CO2 Model problems using graphs and solve these problems algorithmically.
CO3 Illustrate fundamentals of spanning tree, circuits and cut-sets.
CO4 Apply this knowledge in (especially) computer science applications.

Unit	Contents	Hours Allotted
1	**Unit 1: Graphs and operations on graphs** Definition and elementary results Types of graphs Isomorphism Matrix representation of graphs: Adjacency matrix and incidence matrix Subgraphs and induced graphs Complement of a graph, Self complementary graphs Union, intersection of graphs, Ring sum of two graphs Definitions: walk, trail, tour, path and circuit, Definitions of connected, disconnected graphs Dijkstra's shortest path algorithm Connectivity: Isthumus, cut-vertex, Edge connectivity & vertex connectivity.	15
2	**Unit 2: Tree Graphs** Tree: Definition Properties of Trees: Theorem: A tree with n vertices has n -1 edges. Theorem: A connected graph G with n vertices and n - 1 edges is a tree Theorem: A graph with n vertices is a tree if and only if it is circuit freeand has n - 1 edges. Theorem: A graph G is a tree if and only if it is minimally connected. Centre of a tree Spanning tree: Definition and examples Fundamental circuit and cut-set: Definition, examples. Binary trees and elementary results, examples. Kruskal's algorithm, examples.	15

Reference Books

1. Discrete Mathematics by Kenneth Rosen, Tata McGraw Hill
2. Graph Theory with Applications to Computer Sc. & Engg. by Narsing Deo, PHI, 2009
3. A First Step in Graph Theory by Raghunathan, Nimkar and Solapurrkar
4. Discrete mathematics by S.R.Patil and others, NIRALI Prakashan.
5. Discrete mathematics by Bhopatkar, Nimbkar, Joglekar, VISION Publication.

6. Introduction to Graph theory by S. Arumugham and S. Ramachandran, published by Scitech Publications, Chennai-17
7. Introduction to Graph Theory, Mamta Chaudhary, Vani Sharma and Pooja Yadav, Sultan Chand & Sons, Educational Publishers, New Delhi.

Course Code: **Subject II DSC IV**: Course Title**: Group and Coding Theory**

Total Contact Hours: **30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Learn Group structure and its properties.

CO2 Understand fundamental properties of sub-groups, cyclic groups, permutation groups.

CO3 identify different types of group structure and apply them in Cryptography

CO4 Compile the concepts, properties, aspects of Algebra and apply them in computer science.

Unit	Contents	Hours Allotted
1	**Unit 1: Groups** Binary Operation Group: Definition and Examples Elementary Properties of Groups Order of a group, order of an element Examples (Zn, +) and (U(n), *) Subgroup definition, Finite subgroup test, subgroups of Zn Generator, cyclic group, finding generators of Zn (Corollary without proof)	15
2	**Unit 2: Permutation group and Coding Theory** Permutation group, definition, composition of two permutations, representationas product of disjoint cycles, inverse and order of a permutation, even / odd permutations. Cosets: Definition, Examples and Properties, Lagrange Theorem (without Proof) Definitions: Ring, Integral domain, Field. Coding of Binary Information and Error detection Decoding and Error Correction Public Key Cryptography	15

Reference Books

1. Groups and Coding theory by Kalyanrao Takale and others, Nirali Prakashan (Golden series), 2020.
2. Groups and Coding theory by Parshuram Ahire, Vision Publications, 2020.
3. Contemporary Abstract Algebra by J. A, Gallian (Eighth Edition), Cengage Learning India Private Limited, Delhi. Fourth impression, 2015.
4. Discrete Mathematical Structures by Bernard Kolman, Robert C. Busby and Sharon Ross (6^{th} Edition) Pearson Education Publication.

Course Code: **Subject II Practical II**: Course Title: **Mathematics Laboratory course–II**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

List of Practical's:

Pr. No	**Title of the Practical**	**No. of Practical**
1.	Union, intersection & Ring sum of two graphs	1
2.	Dijkstra's Shortest path algorithm	1
3.	Fundamental circuit and fundamental cut set	1
4.	Kruskal's algorithm	1
5.	Examples on Group	1
6.	Examples on subgroup and finite subgroup test	1
7.	Examples on finding generators and subgroups of Zn	1
8.	Examples on permutations (order, inverse, even/odd)	1
9.	Examples on parity check matrix and group code	1
10.	C – Program to check whether given number has odd parity or even parity	1
11.	C – program to convert given input string into cipher text	1
12.	C – program to Convert given graph into matrix form	1

Course Code: **Subject III DSC III**: Course Title**: Sensors and Signal Conditioning**

Total Contact Hours: **30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

CO1: On completion of the course, the students will be able to:CO1: After completion of this course, student will be able to understand the sensors.

CO2: Describe the working principle, selection criteria and applications of various transducers used in the instrumentation systems.

CO3: Getting a knowledge of signal conditioning circuits, data converters & digital instruments..

CO4: Understanding of different Actuators, Data Acquisition Systems & Data loggers.

Unit	Contents	Hours Allotted
1	**A) Sensors & Transducers:** Definition of Transducer & Sensor, Classification of Transducers & Sensors, Characteristics of Transducers, Specifications of Transducers (Accuracy, Range, Linearity, Sensitivity, Resolution, Reproducibility), Temperature: Thermocouple, RTD, LM35, Thermistors, Pressure/ Force: Strain-Gauge, Piezo-Electric, LVDT, Capacitive Transducers, Optical: LDR, Photovoltaic Cell, Proximity: Hall effect sensor, Ultrasonic sensor, PIR (passive Infrared sensor), **B) Signal Conditioning & Data Convertors:** Introduction, Signal conditioning of passive sensors using Wheatstone's bridge, Pre-Amplifiers, **Filters**: Concept, Active filters, Passive Filters **Digital Signal conditioning**: Types of **ADC:** SAR-ADC , Flash-ADC, Specifications of ADC (Linearity, Resolution, Conversion time, Accuracy), Types of **DAC:** Binary, weighted resistors, R-2R Ladder DAC, Specifications of DAC (Linearity, Resolution, Accuracy),	15
2	**A) Actuators & Data Acquisition Systems:** Definition of Actuators, Types of Actuators, Electrical Actuators: Relays, Motors: AC, DC, Servo, Stepper, **Data Acquisition System**s: Generalized DAS system, Signal conditioning for DAS, Types of DAS systems, Multiplexing, Sample and Hold Circuit, Computer based DAS system, Data Loggers. **B) Digital Instruments & Display devices:** Digital Multi-meter, Digital Frequency Meter, Digital Universal Counter, Digital Tachometer, Digital Phase Meter, Concept of Digital Storage Oscilloscope, Digital Displays: LCD, LED, OLED Displays. (Comparative study),	15

Reference Books

1. Electronic Instrumentation: by Kalsi, TMH
2. Transducers & Instrumentation: by Murthy PHI (Unit1)
3. Instrumentation Measurements & Analysis: by Nakra & Chaudhary TMH
4. Instrumentation Devices & Systems: by Rangan, Sharma, Mani, TMH

Course Code: **Subject III DSC IV**: Course Title**: Advanced Digital Electronics**

Total Contact Hours: **30 hrs. (30 lectures of 60 min)**

Credits**: 02** Teaching Scheme**: Theory – 2 Lectures / Week** Total Marks**: 50**

Course Outcomes (COs):

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

CO1 Understand the Sequential Circuits like Flip-Flop,
CO2 Understand the various digital Counters & Shift registers,
CO3 Understand 8-bit Microprocessor-8085 architecture,
CO4 Understand 8085-microprocessor Instruction set & assembly language programming.

Unit	Contents	Hours Allotted
1	**A) Sequential Circuits:** Concept of Sequential circuits **:** types of **Flip-flops**: RS flip-flop (NAND & NOR), Clocked RS flip-flop, D flip-flop, Edge-triggering & Level Triggering, JK flip-flop, Master-Slave JK flip-flop, PRESET & CLEAR inputs in a Flip-flop, T-Flipflop, Conversion of one Flip-flop into another flip-flop. **B) Counters & Shift Registers:** Types of counters, Binary Counter, Decade Counter, Asynchronous (Ripple counter), Synchronous counter, 4-bit Ripple Counter, Up-Down counter (3-bit), Modulus-N counter, Construction of Mod-5, Mod-10 counter, IC-7490, Shift Registers: Operating modes - SISO, SIPO, PISO, PIPO, Applications: Ring Counter, Johnson Counter, IC-7495,	15
2	**A) Semiconductor Memories:** Classification of memories, Block diagram of memory device, Memory Read & Memory Write operation, Static RAM, Dynamic RAM, ROM, EPROM, EEPROM, Flash memory, Comparison: Static RAM vs Dynamic RAM, EEPROM vs Flash memory, Characteristics of memory, **B) Introduction to Microprocessors:** General block diagram of CPU, Introduction & evolution of Microprocessors (4, 8, 16, 32Bits), 8-bit Microprocessor (Intel 8085): Pin Diagram of IC-8085, Features of IC-8085, Internal Architecture of IC-8085, Instruction Set of IC-8085. Programming : types of Instructions, Instruction format, addressing modes, **Assembly language programming**: for Data transfer, Addition, Subtraction, Multiplication, Division, Memory Block Transfer & Block Exchange operations, shifting of the bits.	15

Reference Books:

1. Microprocessors-8085: by Ramesh Gaonkar,
2. Microprocessor -8085: by Vibhute & Borule,
3. Digital Principles and Applications: by Malvino Leach, Tata McGraw Hill.
4. Fundamentals of Digital Electronics: by Anand Kumar, PHI Publication.
5. Digital Electronics: by R. P Jain,

Course Code: **Subject III Practical II**:

Course Title: **Practical Based on Subject III DSC-III and Subject III DSC-IV**

Credits**: 02** Teaching Scheme**: Practical's – 4 Lectures / Week / batch** Total Marks**: 50**

List of Practical's:

Sr. No.	Titles of the Practicals
1	Study of R-2R Ladder DAC.
2	Study of Instrumentation amplifier by using three Op-Amps.
3	Study of ON-OFF temperature Controller by using LM35/RTD/Thermocouple.
4	Study of Asynchronous Up or Down counter.
5	Study of Universal Shift register.
6	Study of Ring Counter & Johnson Counter,
7	Write ALP for Addition of two 8 bit numbers.
8	Write ALP for Addition of two 16 bit numbers.
9	Write ALP for Subtraction of two 8 bit numbers.
10	Write ALP for Subtraction of two 16 bit numbers.
11	Write ALP for Multiplication of two 8 bit numbers.
12	Write ALP for Division of two 8 bit numbers.
13	Write ALP for Memory block transfer or Memory block exchange.
14	Write ALP to find the largest number or smallest number.
15	Write ALP to find Odd number or Even number.
16	Write ALP to find 1's complement of 8 bit number.
17	Write ALP to find 1's complement of 16 bit number.
18	Write ALP to find 2's complement of 8 bit number.
19	Write ALP to shift 8 bit number to the left by 1 bit & 2 bits.
20	Write ALP to shift 8 bit number to the right by 1 bit & 2 bits.

Course Name: Business Statistics using MS Excel/Linux Practical – II Practical: 60 Hrs. Marks: 50 (Credits: 02)

Prerequisite:

The student opting this course must have opted Statistics using MS Excel/Linux Practical-Icourse in semester I

Course Outcomes:

After completion of this practical course, the student will be able to:

i. get the basic knowledge of bivariate data analysis by computing correlationcoefficient and performing linear regression analysis.
ii. get the knowledge of discrete probability distributions.
iii. implement the probability distribution concepts using model sampling.
iv. acquire the insights of time series and index number theories with its application.

List of Practical's:

1. Computation of Correlation Coefficient and Scatter Diagram (ungrouped Data)
2. Fitting of linear and non-linear regression: Obtaining Linear Regression Equations and estimation of dependent variable using least square method when bivariate data is given. (ungrouped data)
Computation of index numbers by using i) Simple Method – Aggregative and Relative
ii) Weighted Method – Aggregative and Relative
Laspeyre's and Paasche's method
iii) Fisher's ideal method
4. Computation of trend by using i) Moving average method
ii) Progressive average method
iii) Least square method
5. Fitting of discrete Uniform Distribution
6. Fitting of Binomial distribution
7. Model sampling from Binomial distribution
8. Fitting of Poisson Distribution
9. Model sampling from Poisson Distribution
10. Case study of at least two of first four practical's using the secondary data obtained fromgovernment sites

Note:

- Test of goodness of fit is not necessary for the practical of fitting of distribution.
- All practicals should conducted on Computer using MS Excel/Libre Office Calc(Linux) software.
- Computer printout of each practical with output should be attached to the journal.
- Student must complete the entire practical to the satisfaction of the teacher concerned.
- Student must produce the laboratory journal along with the completion certificatesigned by the Head of the department, at the time of practical examination.

Reference Books:

1. Agarwal B. L. (2019) *Basic Statistics*, New Age International (P) Limited.
2. Gupta S. C. (2019) *Fundamentals of Statistics*, Himalaya Publishing House Pvt. Ltd.
3. *A First Course in Probability* by Sheldon Ross (2022), Pearson pub.
4. *Statistical Methods* (An introductory text by J. Medhi), New Age International (P)Limited.
5. *Business Statistics: A First Course* by David Levine, Katherene szabat, Pearson Pub.
6. Sharma V. K. (2012) *Elements of Statistics*, Gullybaba Publishing House Pvt. Ltd.

Practical Examination:

1. Practical Examination will be conducted at the end of each Semester.
2. Each practical paper carries 50 Marks.
3. Duration of Practical Examination: 4 Hrs.

Nature of Question Paper:

i. There will be four questions of 18 marks each.
ii. In each question there are two sub questions (a) and (b) each carrying 09 marks
iii. Students have to attempt any two out of four questions.
iv. The distribution of practical paper's marks:

- Two questions each of 18 marks (Total 18 x 02 = 36 Marks)
- Certified Journal: 05 Marks,
- Viva voce: 04 Marks
- Case study: 05 marks
- Total Marks: 50

Course Code**: VEC- I** Course Title: **Democracy, Election and Constitution**	Syllabus will be provided by Shivaji University as per NEP