

Mathematics for Computer Science		Semester	3
Course Code	BCS301	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 20 Hours Tutorial	Total Marks	100
Credits	04	Exam Hours	3
Examination type (SEE)	Theory		
Course objectives: This course will enable the students to: 1. To introduce the concept of random variables, probability distributions, specific discrete and continuous distributions with practical application in Computer Science Engineering and social life situations. 2. To Provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypotheses. 3. To Determine whether an input has a statistically significant effect on the system's response through ANOVA testing.			
Teaching-Learning Process Pedagogy (General Instructions): Teachers can use the following strategies to accelerate the attainment of the various course outcomes. 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. 2. State the need for Mathematics with Engineering Studies and Provide real-life examples. 3. Support and guide the students for self-study. 4. You will assign homework, grading assignments and quizzes, and documenting students' progress. 5. Encourage the students to group learning to improve their creative and analytical skills. 6. Show short related video lectures in the following ways: <ul style="list-style-type: none">• As an introduction to new topics (pre-lecture activity).• As a revision of topics (post-lecture activity).• As additional examples (post-lecture activity).• As an additional material of challenging topics (pre-and post-lecture activity).• As a model solution of some exercises (post-lecture activity).			
Module-1: Probability Distributions			
Probability Distributions: Review of basic probability theory. Random variables (discrete and continuous), probability mass and density functions. Mathematical expectation, mean and variance. Binomial, Poisson and normal distributions- problems (derivations for mean and standard deviation for Binomial and Poisson distributions only)-Illustrative examples. Exponential distribution. (12 Hours) (RBT Levels: L1, L2 and L3)			
Pedagogy	Chalk and Board, Problem-based learning		
Module-2: Joint probability distribution & Markov Chain			

Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance and correlation. Markov Chain: Introduction to Stochastic Process, Probability Vectors, Stochastic matrices, Regular stochastic matrices, Markov chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states. (12 Hours) (RBT Levels: L1, L2 and L3)	
Pedagogy	Chalk and Board, Problem-based learning
Module-3: Statistical Inference 1	
Introduction, sampling distribution, standard error, testing of hypothesis, levels of significance, test of significances, confidence limits, simple sampling of attributes, test of significance for large samples, comparison of large samples. (12 Hours) (RBT Levels: L1, L2 and L3)	
Pedagogy	Chalk and Board, Problem-based learning
Module-4: Statistical Inference 2	
Sampling variables, central limit theorem and confidences limit for unknown mean. Test of Significance for means of two small samples, students 't' distribution, Chi-square distribution as a test of goodness of fit. F-Distribution. (12 Hours) (RBT Levels: L1, L2 and L3)	
Pedagogy	Chalk and Board, Problem-based learning
Module-5: Design of Experiments & ANOVA	
Principles of experimentation in design, Analysis of completely randomized design, randomized block design. The ANOVA Technique, Basic Principle of ANOVA, One-way ANOVA, Two-way ANOVA, Latin-square Design, and Analysis of Co-Variance. (12 Hours) (RBT Levels: L1, L2 and L3)	
Pedagogy	Chalk and Board, Problem-based learning
Course outcome (Course Skill Set) At the end of the course, the student will be able to: <ol style="list-style-type: none"> 1. Explain the basic concepts of probability, random variables, probability distribution 2. Apply suitable probability distribution models for the given scenario. 3. Apply the notion of a discrete-time Markov chain and n-step transition probabilities to solve the given problem 4. Use statistical methodology and tools in the engineering problem-solving process. 5. Compute the confidence intervals for the mean of the population. 6. Apply the ANOVA test related to engineering problems. 	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment

Test component, there are 25 marks.

- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**).

1. The question paper will have ten questions. Each question is set for 20 marks.
 2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
 3. The students have to answer 5 full questions, selecting one full question from each module.
- Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:**Textbooks:**

1. **Ronald E. Walpole, Raymond H Myers, Sharon L Myers & Keying Ye** “Probability & Statistics for Engineers & Scientists”, Pearson Education, 9th edition, 2017.
2. **Peter Bruce, Andrew Bruce & Peter Gedeck** “Practical Statistics for Data Scientists” O’Reilly Media, Inc., 2nd edition **2020**.

Reference Books: (Name of the author/Title of the Book/ Name of the publisher/Edition and Year)

1. **Erwin Kreyszig**, “Advanced Engineering Mathematics”, John Wiley & Sons, 9th Edition, 2006.
2. **B. S. Grewal** “Higher Engineering Mathematics”, Khanna publishers, 44th Ed., 2021.
3. **G Haribaskaran** “Probability, Queuing Theory & Reliability Engineering”, Laxmi Publication, Latest Edition, 2006
4. **Irwin Miller & Marylees Miller**, John E. Freund’s “Mathematical Statistics with Applications” Pearson. Dorling Kindersley Pvt. Ltd. India, 8th edition, 2014.
5. **S C Gupta and V K Kapoor**, “Fundamentals of Mathematical Statistics”, S Chand and Company, Latest edition.
6. **Robert V. Hogg, Joseph W. McKean & Allen T. Craig**. “Introduction to Mathematical Statistics”, Pearson Education 7th edition, 2013.
7. **Jim Pitman**. Probability, Springer-Verlag, 1993.
8. **Sheldon M. Ross**, “Introduction to Probability Models” 11th edition. Elsevier, 2014.
9. **A. M. Yaglom and I. M. Yaglom**, “Probability and Information”. D. Reidel Publishing Company. Distributed by Hindustan Publishing Corporation (India) Delhi, 1983.
10. **P. G. Hoel, S. C. Port and C. J. Stone**, “Introduction to Probability Theory”, Universal Book Stall, (Reprint), 2003.
11. **S. Ross**, “A First Course in Probability”, Pearson Education India, 6th Ed., 2002.
12. **W. Feller**, “An Introduction to Probability Theory and its Applications”, Vol. 1, Wiley, 3rd Ed., 1968.
13. **N.P. Bali and Manish Goyal**, A Textbook of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
14. **Veerarajan T**, Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010

Web links and Video Lectures (e-Resources):

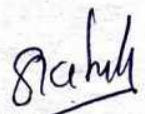
<http://nptel.ac.in/courses.php?disciplineID=111>
[http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
<http://academicearth.org/> <http://www.bookstreet.in>.

VTU EDUSAT PROGRAMME – 20

VTU e-Shikshana Program

Activity-Based Learning (Suggested Activities in Class)/Practical-Based Learning

- Programming Assignment
- Seminars


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar. MOODBIDRI - 574 225

Digital Design and Computer Organization		Semester	3
Course Code	BCS302	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 20 Hours of Practicals	Total Marks	100
Credits	04	Exam Hours	3
Examination nature (SEE)	Theory		
Course objectives: <ul style="list-style-type: none">• To demonstrate the functionalities of binary logic system• To explain the working of combinational and sequential logic system• To realize the basic structure of computer system• To illustrate the working of I/O operations and processing unit			
Teaching-Learning Process (General Instructions) <p>These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes. 1. Chalk and Talk</p> <p>2. Live Demo with experiments</p> <p>3. Power point presentation</p>			
MODULE-1		8 Hr	
Introduction to Digital Design: Binary Logic, Basic Theorems And Properties Of Boolean Algebra, Boolean Functions, Digital Logic Gates, Introduction, The Map Method, Four-Variable Map, Don't-Care Conditions, NAND and NOR Implementation, Other Hardware Description Language – Verilog Model of a simple circuit.			
Text book 1: 1.9, 2.4, 2.5, 2.8, 3.1, 3.2, 3.3, 3.5, 3.6, 3.9			
MODULE-2		8 Hr	
Combinational Logic: Introduction, Combinational Circuits, Design Procedure, Binary Adder- Subtractor, Decoders, Encoders, Multiplexers. HDL Models of Combinational Circuits – Adder, Multiplexer, Encoder.			
Sequential Logic: Introduction, Sequential Circuits, Storage Elements: Latches, Flip-Flops.			
Text book 1: 4.1, 4.2, 4.4, 4.5, 4.9, 4.10, 4.11, 4.12, 5.1, 5.2, 5.3, 5.4.			
MODULE-3		8 Hr	
Basic Structure of Computers: Functional Units, Basic Operational Concepts, Bus structure, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.			
Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instruction and Instruction sequencing, Addressing Modes.			
Text book 2: 1.2, 1.3, 1.4, 1.6, 2.2, 2.3, 2.4, 2.5			
MODULE-4		8 Hr	

Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access: Bus Arbitration, Speed, size and Cost of memory systems. Cache Memories – Mapping Functions.

Text book 2: 4.1, 4.2.1, 4.2.2, 4.2.3, 4.4, 5.4, 5.5.1

MODULE-5

8 Hr

Basic Processing Unit: Some Fundamental Concepts: Register Transfers, Performing ALU operations, fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction. **Pipelining:** Basic concepts, Role of Cache memory, Pipeline Performance.

Text book 2: 7.1, 7.2, 8.1

PRACTICAL COMPONENT OF IPCC

Sl.N O	Experiments
	Simulation packages preferred: Multisim, Modelsim, PSpice or any other relevant
1	Given a 4-variable logic expression, simplify it using appropriate technique and simulate the same using basic gates.
2	Design a 4 bit full adder and subtractor and simulate the same using basic gates.
3	Design Verilog HDL to implement simple circuits using structural, Data flow and Behavioural model.
4	Design Verilog HDL to implement Binary Adder-Subtractor – Half and Full Adder, Half and Full Subtractor.
5	Design Verilog HDL to implement Decimal adder.
6	Design Verilog program to implement Different types of multiplexer like 2:1, 4:1 and 8:1.
7	Design Verilog program to implement types of De-Multiplexer.
8	Design Verilog program for implementing various types of Flip-Flops such as SR, JK and D.

Course outcomes (Course Skill Set):

At the end of the course, the student will be able to:

CO1: Apply the K-Map techniques to simplify various Boolean expressions.

CO2: Design different types of combinational and sequential circuits along with Verilog programs.

CO3: Describe the fundamentals of machine instructions, addressing modes and Processor performance.

CO4: Explain the approaches involved in achieving communication between processor and I/O devices.

CO5: Analyze internal Organization of Memory and Impact of cache/Pipelining on Processor Performance.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other

assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.

- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks**).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC. **CIE for the practical component of the IPCC**
- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test (**duration 02/03 hours**) after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks**.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources:

Books

1. M. Morris Mano & Michael D. Ciletti, Digital Design With an Introduction to Verilog Design, 5e, Pearson Education.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, 5th Edition, Tata McGraw Hill.

Web links and Video Lectures (e-Resources):


<https://cse11-iiith.vlabs.ac.in/>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Assign the group task to Design the various types of counters and display the output accordingly

Assessment Methods

- Lab Assessment (25 Marks)
- GATE Based Aptitude Test


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar, MOODBIDRI - 574 225

OPERATING SYSTEMS		Semester	3
Course Code	BCS303	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 20 hours practicals	Total Marks	100
Credits	04	Exam Hours	3
Examination nature (SEE)	Theory		
Course objectives: <ul style="list-style-type: none">• To Demonstrate the need for OS and different types of OS• To discuss suitable techniques for management of different resources• To demonstrate different APIs/Commands related to processor, memory, storage and file system management.			
Teaching-Learning Process (General Instructions) <p>Teachers can use the following strategies to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none">1. Lecturer methods (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.2. Use of Video/Animation to explain functioning of various concepts.3. Encourage collaborative (Group Learning) Learning in the class.4. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.5. Role play for process scheduling.6. Demonstrate the installation of any one Linux OS on VMware/Virtual Box			
MODULE-1		8 Hours	
Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments.			
Operating System Services: User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System debugging, Operating System generation; System boot.			
Textbook 1: Chapter – 1 (1.1-1.12), 2 (2.2-2.11)			
MODULE-2		8 Hours	

<p>Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication</p> <p>Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.</p> <p>Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Thread scheduling; Multiple-processor scheduling,</p> <p>Textbook 1: Chapter – 3 (3.1-3.4), 4 (4.1-4.4), 5 (5.1 -5.5)</p>	
MODULE-3	8 Hours
<p>Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization;</p> <p>Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.</p> <p>Textbook 1: Chapter – 6 (6.1-6.6), 7 (7.1 -7.7)</p>	
MODULE-4	8 Hours
<p>Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.</p> <p>Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.</p> <p>Textbook 1: Chapter -8 (8.1-8.6), 9 (9.1-9.6)</p>	
MODULE-5	8 Hours
<p>File System, Implementation of File System: File system: File concept; Access methods; Directory and Disk structure; File system mounting; File sharing; Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.</p> <p>Secondary Storage Structure, Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix.</p> <p>Textbook 1: Chapter – 10 (10.1-10.5) ,11 (11.1-11.5),12 (12.1-12.5), 14 (14.1-14.4)</p>	

PRACTICAL COMPONENT OF IPCC *(May cover all / major modules)*

SLN O	Experiments
1	Develop a c program to implement the Process system calls (fork (), exec(), wait(), create process, terminate process)

2	Simulate the following CPU scheduling algorithms to find turnaround time and waiting time a) FCFS b) SJF c) Round Robin d) Priority.
3	Develop a C program to simulate producer-consumer problem using semaphores.
4	Develop a C program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.
5	Develop a C program to simulate Bankers Algorithm for DeadLock Avoidance.
6	Develop a C program to simulate the following contiguous memory allocation Techniques: a) Worst fit b) Best fit c) First fit.
7	Develop a C program to simulate page replacement algorithms: a) FIFO b) LRU
8	Simulate following File Organization Techniques a) Single level directory b) Two level directory
9	Develop a C program to simulate the Linked file allocation strategies.
10	Develop a C program to simulate SCAN disk scheduling algorithm.

Course outcomes (Course Skill Set):

At the end of the course, the student will be able to:

CO 1. Explain the structure and functionality of operating system

CO 2. Apply appropriate CPU scheduling algorithms for the given problem.

CO 3. Analyse the various techniques for process synchronization and deadlock handling.

CO 4. Apply the various techniques for memory management
CO 5. Explain file and secondary storage management strategies.

CO 6. Describe the need for information protection mechanisms

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods

mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.

- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks**). ● The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC. **CIE for the practical component of the IPCC**
- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test (**duration 02/03 hours**) after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks**.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**. ● The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources:

Textbooks

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2015

Reference Books


1. Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6th Edition
2. D.M Dhamdhare, Operating Systems: A Concept Based Approach 3rd Ed, McGraw- Hill, 2013.
3. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE), 2014.
4. William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson.

Web links and Video Lectures (e-Resources):

1. <https://youtu.be/mXw9ruZaxzQ>
2. <https://youtu.be/vBURt97EkA>
3. https://www.youtube.com/watch?v=783KAB-tuE4&list=PLIemF3uozcAKTgsCIj82voMK3TMR0YE_f
4. <https://www.youtube.com/watch?v=3-ITLMMeeXY&list=PL3pGy4HtqwD0n7bQfHjPnsWzkeRn6mkO>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Assessment Methods ○ Case Study on Unix Based Systems (10 Marks) ○ Lab Assessment (25 Marks)


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar, MOODBIDRI - 574 225

DATA STRUCTURES AND APPLICATIONS		Semester	3
Course Code	BCS304	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	3
Examination type (SEE)	Theory		
Course objectives: CLO 1. To explain fundamentals of data structures and their applications. CLO 2. To illustrate representation of Different data structures such as Stack, Queues, Linked Lists, Trees and Graphs. CLO 3. To Design and Develop Solutions to problems using Linear Data Structures CLO 4. To discuss applications of Nonlinear Data Structures in problem solving. CLO 5. To introduce advanced Data structure concepts such as Hashing and Optimal Binary Search Trees			
Teaching-Learning Process (General Instructions) Teachers can use following strategies to accelerate the attainment of the various course outcomes. 1. Chalk and Talk with Black Board 2. ICT based Teaching 3. Demonstration based Teaching			
Module-1		8Hours	
INTRODUCTION TO DATA STRUCTURES: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations Review of pointers and dynamic Memory Allocation, ARRAYS and STRUCTURES: Arrays, Dynamic Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings STACKS: Stacks, Stacks Using Dynamic Arrays, Evaluation and conversion of Expressions Text Book: Chapter-1:1.2 Chapter-2: 2.1 to 2.7 Chapter-3: 3.1,3.2,3.6 Reference Book 1: 1.1 to 1.4			
Module-2		8Hours	
QUEUES: Queues, Circular Queues, Using Dynamic Arrays, Multiple Stacks and queues. LINKED LISTS : Singly Linked, Lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials Text Book: Chapter-3: 3.3, 3.4, 3.7 Chapter-4: 4.1 to 4.4			
Module-3		8Hours	
LINKED LISTS : Additional List Operations, Sparse Matrices, Doubly Linked List. TREES: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees. Text Book: Chapter-4: 4.5,4.7,4.8 Chapter-5: 5.1 to 5.3, 5.5			
Module-4		8Hours	

	TREES(Cont.): Binary Search trees, Selection Trees, Forests, Representation of Disjoint sets, Counting Binary Trees, GRAPHS: The Graph Abstract Data Types, Elementary Graph Operations Text Book: Chapter-5: 5.7 to 5.11 Chapter-6: 6.1, 6.2
	Module-5 8Hours

	HASHING: Introduction, Static Hashing, Dynamic Hashing PRIORITY QUEUES: Single and double ended Priority Queues, Leftist Trees INTRODUCTION TO EFFICIENT BINARY SEARCH TREES: Optimal Binary Search Trees Text Book: Chapter 8: 8.1 to 8.3 Chapter 9: 9.1, 9.2 Chapter 10: 10.1
	Course outcome (Course Skill Set) At the end of the course the student will be able to: CO 1. Explain different data structures and their applications. CO 2. Apply Arrays, Stacks and Queue data structures to solve the given problems. CO 3. Use the concept of linked list in problem solving. CO 4. Develop solutions using trees and graphs to model the real-world problem. CO 5. Explain the advanced Data Structures concepts such as Hashing Techniques and Optimal Binary Search Trees.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 220B2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:**Textbook:**

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014

Reference Books:

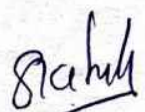
1. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning, 2014.
3. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
4. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013
5. A M Tenenbaum, Data Structures using C, PHI, 1989
6. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

Web links and Video Lectures (e-Resources):

- <http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html>
- <https://nptel.ac.in/courses/106/105/106105171/>
- <http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html>
- https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s
- <https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html>
- <https://nptel.ac.in/courses/106/102/106102064/>
- <https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html>
- <https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html>
- <https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html>
- <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html>
- <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html>
- https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01350159542807756812559/overview

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Role Play
- Flipped classroom
- Assessment Methods for 25 Marks (opt two Learning Activities) ○ Case Study ○ Programming Assignment ○ Gate Based Aptitude Test
 - MOOC Assignment for selected Module


 HOD's Signature
H.O.D.
 Dept. Of Information Science & Engineering
 Alva's Institute of Engg. & Technology
 Miiar, MOODBIDRI - 574 225

DATA STRUCTURES LABORATORY SEMESTER – III			
Course Code	BCSL305	CIE Marks	50
Number of Contact Hours/Week	0:0:2	SEE Marks	50
Total Number of Lab Contact Hours	28	Exam Hours	03
Credits – 1			
Course Learning Objectives:			
<p>This laboratory course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of</p> <ul style="list-style-type: none"> • Dynamic memory management • Linear data structures and their applications such as stacks, queues and lists • Non-Linear data structures and their applications such as trees and graphs 			
Descriptions (if any):			
<ul style="list-style-type: none"> • Implement all the programs in “C ” Programming Language and Linux OS. 			
Programs List:			
1.	<p>Develop a Program in C for the following:</p> <ol style="list-style-type: none"> Declare a calendar as an array of 7 elements (A dynamically Created array) to represent 7 days of a week. Each Element of the array is a structure having three fields. The first field is the name of the Day (A dynamically allocated String), The second field is the date of the Day (A integer), the third field is the description of the activity for a particular day (A dynamically allocated String). Write functions create(), read() and display(); to create the calendar, to read the data from the keyboard and to print weeks activity details report on screen. 		
2.	<p>Develop a Program in C for the following operations on Strings.</p> <ol style="list-style-type: none"> Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR <p>Support the program with functions for each of the above operations. Don't use Built-in functions.</p>		


3.	<p>Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)</p> <ol style="list-style-type: none"> Push an Element on to Stack Pop an Element from Stack Demonstrate how Stack can be used to check Palindrome Demonstrate Overflow and Underflow situations on Stack Display the status of Stack Exit <p>Support the program with appropriate functions for each of the above operations</p>
4.	<p>Develop a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands.</p>
5.	<p>Develop a Program in C for the following Stack Applications</p> <ol style="list-style-type: none"> Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ Solving Tower of Hanoi problem with n disks
6.	<p>Develop a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)</p> <ol style="list-style-type: none"> Insert an Element on to Circular QUEUE Delete an Element from Circular QUEUE Demonstrate Overflow and Underflow situations on Circular QUEUE Display the status of Circular QUEUE Exit <p>Support the program with appropriate functions for each of the above operations</p>
7.	<p>Develop a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: <i>USN, Name, Programme, Sem, PhNo</i></p> <ol style="list-style-type: none"> Create a SLL of N Students Data by using <i>front insertion</i>. Display the status of SLL and count the number of nodes in it Perform Insertion / Deletion at End of SLL Perform Insertion / Deletion at Front of SLL (Demonstration of stack) Exit
8.	<p>Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: <i>SSN, Name, Dept, Designation, Sal, PhNo</i></p> <ol style="list-style-type: none"> Create a DLL of N Employees Data by using <i>end insertion</i>. Display the status of DLL and count the number of nodes in it Perform Insertion and Deletion at End of DLL Perform Insertion and Deletion at Front of DLL Demonstrate how this DLL can be used as Double Ended Queue. Exit

9.	<p>Develop a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes</p> <ol style="list-style-type: none"> Represent and Evaluate a Polynomial $P(x,y,z) = 6x^2y^2z - 4yz^5 + 3x^3yz + 2xy^5z - 2xyz^3$ Find the sum of two polynomials $POLY1(x,y,z)$ and $POLY2(x,y,z)$ and store the result in $POLYSUM(x,y,z)$ <p>Support the program with appropriate functions for each of the above operations</p>
10.	<p>Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers .</p> <ol style="list-style-type: none"> Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 Traverse the BST in Inorder, Preorder and Post Order Search the BST for a given element (KEY) and report the appropriate message Exit
11.	<p>Develop a Program in C for the following operations on Graph(G) of Cities</p> <ol style="list-style-type: none"> Create a Graph of N cities using Adjacency Matrix. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method
12.	<p>Given a File of N employee records with a set K of Keys (4-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table (HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers. Develop a Program in C that uses Hash function H:</p> <p>$K \rightarrow L$ as $H(K) = K \bmod m$ (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.</p>
Laboratory Outcomes: The student should be able to:	

- Analyze various linear and non-linear data structures
- Demonstrate the working nature of different types of data structures and their applications
- Use appropriate searching and sorting algorithms for the give scenario.
- Apply the appropriate data structure for solving real world problems

Conduct of Practical Examination:

- Experiment distribution
 - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
 - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Need to change in accordance with university regulations*)
 - For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
 - For laboratories having PART A and PART B
 - Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
 - Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miar, MOODBIDRI - 574 225

Object Oriented Programming with JAVA		Semester	3
Course Code	BCS306A	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	2:0:2	SEE Marks	50
Total Hours of Pedagogy	28 Hours of Theory + 20 Hours of Practical	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
Note - Students who have undergone “ Basics of Java Programming-BPLCK105C/205C” in first year are not eligible to opt this course			
Course objectives: <ul style="list-style-type: none">To learn primitive constructs JAVA programming language.To understand Object Oriented Programming Features of JAVA.To gain knowledge on: packages, multithreaded programing and exceptions.			
Teaching-Learning Process (General Instructions) <p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes and make Teaching –Learning more effective</p> <ol style="list-style-type: none">Use Online Java Compiler IDE: https://www.jdoodle.com/online-java-compiler/ or any other.Demonstration of programing examples.Chalk and board, power point presentationsOnline material (Tutorials) and video lectures.			
Module-1			
An Overview of Java: Object-Oriented Programming (Two Paradigms, Abstraction, The Three OOP Principles), Using Blocks of Code, Lexical Issues (Whitespace, Identifiers, Literals, Comments, Separators, The Java Keywords). Data Types, Variables, and Arrays: The Primitive Types (Integers, Floating-Point Types, Characters, Booleans), Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, Introducing Type Inference with Local Variables. Operators: Arithmetic Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses. Control Statements: Java’s Selection Statements (if, The Traditional switch), Iteration Statements (while, do-while, for, The For-Each Version of the for Loop, Local Variable Type Inference in a for Loop, Nested Loops), Jump Statements (Using break, Using continue, return). Chapter 2, 3, 4, 5			
Module-2			
Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection. Methods and Classes: Overloading Methods, Objects as Parameters, Argument Passing, Returning Objects, Recursion, Access Control, Understanding static, Introducing final, Introducing Nested and Inner Classes. Chapter 6, 7			
Module-3			

	<p>Inheritance: Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Local Variable Type Inference and Inheritance, The Object Class.</p> <p>Interfaces: Interfaces, Default Interface Methods, Use static Methods in an Interface, Private Interface Methods.</p> <p>Chapter 8, 9</p>
	Module-4
	<p>Packages: Packages, Packages and Member Access, Importing Packages.</p> <p>Exceptions: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions.</p> <p>Chapter 9, 10</p>
	Module-5
	<p>Multithreaded Programming: The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads, Obtaining a Thread's State.</p> <p>Enumerations, Type Wrappers and Autoboxing: Enumerations (Enumeration Fundamentals, The values() and valueOf() Methods), Type Wrappers (Character, Boolean, The Numeric Type Wrappers), Autoboxing (Autoboxing and Methods, Autoboxing/Unboxing Occurs in Expressions, Autoboxing/Unboxing Boolean and Character Values).</p> <p>Chapter 11, 12</p>
	<p>Course outcome (Course Skill Set)</p> <p>At the end of the course, the student will be able to:</p> <ol style="list-style-type: none"> 1. Demonstrate proficiency in writing simple programs involving branching and looping structures. 2. Design a class involving data members and methods for the given scenario. 3. Apply the concepts of inheritance and interfaces in solving real world problems. 4. Use the concept of packages and exception handling in solving complex problem 5. Apply concepts of multithreading, autoboxing and enumerations in program development

Programming Experiments (Suggested and are not limited to)

1. Develop a JAVA program to add TWO matrices of suitable order N (The value of N should be read from command line arguments).
2. Develop a stack class to hold a maximum of 10 integers with suitable methods. Develop a JAVA main method to illustrate Stack operations.
3. A class called Employee, which models an employee with an ID, name and salary, is designed as shown in the following class diagram. The method raiseSalary (percent) increases the salary by the given percentage. Develop the Employee class and suitable main method for demonstration.
4. A class called MyPoint, which models a 2D point with x and y coordinates, is designed as follows:
 - Two instance variables x (int) and y (int).
 - A default (or "no-arg") constructor that construct a point at the default location of (0, 0).
 - A overloaded constructor that constructs a point with the given x and y coordinates.
 - A method setXY() to set both x and y.
 - A method getX() which returns the x and y in a 2-element int array.
 - A toString() method that returns a string description of the instance in the format "(x, y)".
 - A method called distance(int x, int y) that returns the distance from this point to another point at the given (x, y) coordinates
 - An overloaded distance(MyPoint another) that returns the distance from this point to the given MyPoint instance (called another)
 - Another overloaded distance() method that returns the distance from this point to the origin (0,0)
 Develop the code for the class MyPoint. Also develop a JAVA program (called TestMyPoint) to test all the methods defined in the class.
5. Develop a JAVA program to create a class named shape. Create three sub classes namely: circle, triangle and square, each class has two member functions named draw () and erase (). Demonstrate

polymorphism concepts by developing suitable methods, defining member data and main program.

6. Develop a JAVA program to create an abstract class Shape with abstract methods calculateArea() and calculatePerimeter(). Create subclasses Circle and Triangle that extend the Shape class and implement the respective methods to calculate the area and perimeter of each shape.
7. Develop a JAVA program to create an interface Resizable with methods resizeWidth(int width) and resizeHeight(int height) that allow an object to be resized. Create a class Rectangle that implements the Resizable interface and implements the resize methods
8. Develop a JAVA program to create an outer class with a function display. Create another class inside the outer class named inner with a function called display and call the two functions in the main class.
9. Develop a JAVA program to raise a custom exception (user defined exception) for DivisionByZero using try, catch, throw and finally.
10. Develop a JAVA program to create a package named mypack and import & implement it in a suitable class.
11. Write a program to illustrate creation of threads using runnable class. (start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).
12. Develop a program to create a class MyThread in this class a constructor, call the base class constructor, using super and start the thread. The run method of the class starts after this. It can be observed that both main thread and created child thread are executed concurrently.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks**).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC. **CIE for the practical component of the IPCC**
- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test (**duration 02/03 hours**) after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks**.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

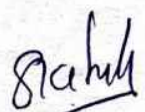
SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources:
Textbook
1. Java: The Complete Reference, Twelfth Edition, by Herbert Schildt, November 2021, McGraw-Hill, ISBN: 9781260463422 Reference Books
1. Programming with Java, 6th Edition, by E Balagurusamy, Mar-2019, McGraw Hill Education, ISBN: 9789353162337. 2. Thinking in Java, Fourth Edition, by Bruce Eckel, Prentice Hall, 2006 (https://sd.blackball.lv/library/thinking_in_java_4th_edition.pdf)
Web links and Video Lectures (e-Resources):
<ul style="list-style-type: none"> • Java Tutorial: https://www.geeksforgeeks.org/java/ • Introduction To Programming In Java (by Evan Jones, Adam Marcus and Eugene Wu): https://ocw.mit.edu/courses/6-092-introduction-to-programming-in-java-january-iap-2010/ • Java Tutorial: https://www.w3schools.com/java/ • Java Tutorial: https://www.javatpoint.com/java-tutorial
Activity Based Learning (Suggested Activities)/ Practical Based learning
1. Installation of Java (Refer: https://www.java.com/en/download/help/index_installing.html) 2. Demonstration of online IDEs like geeksforgeeks, jdoodle or any other Tools 3. Demonstration of class diagrams for the class abstraction, type visibility, composition and inheritance
Assessment Method
● Programming Assignment / Course Project


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar, MOODBIDRI - 574 225

Template for Practical Course and if AEC is a practical Course Annexure-V

R Programming		Semester	3
Course Code	BCS358B	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	02
Examination type (SEE)	Practical		
Course objectives:			
<ul style="list-style-type: none">• To explore and understand how R and R Studio interactive environment.• To understand the different data Structures, data types in R.• To learn and practice programming techniques using R programming.• To import data into R from various data sources and generate visualizations.• To draw insights from datasets using data analytics techniques.			
Sl.NO	Experiments		
1	<p>Demonstrate the steps for installation of R and R Studio. Perform the following:</p> <ul style="list-style-type: none">a) Assign different type of values to variables and display the type of variable. Assign different types such as Double, Integer, Logical, Complex and Character and understand the difference between each data type.b) Demonstrate Arithmetic and Logical Operations with simple examples.c) Demonstrate generation of sequences and creation of vectors.d) Demonstrate Creation of Matricese) Demonstrate the Creation of Matrices from Vectors using Binding Function.f) Demonstrate element extraction from vectors, matrices and arrays <p>Suggested Reading – Text Book 1 – Chapter 1 (What is R, Installing R, Choosing an IDE – RStudio, How to Get Help in R, Installing Extra Related Software), Chapter 2 (Mathematical Operations and Vectors, Assigning Variables, Special Numbers, Logical Vectors), Chapter 3 (Classes, Different Types of Numbers, Other Common Classes, Checking and Changing Classes, Examining Variables)</p>		
2	<p>Assess the Financial Statement of an Organization being supplied with 2 vectors of data: Monthly Revenue and Monthly Expenses for the Financial Year. You can create your own sample data vector for this experiment) Calculate the following financial metrics: a. Profit for each month.</p> <ul style="list-style-type: none">b. Profit after tax for each month (Tax Rate is 30%).c. Profit margin for each month equals to profit after tax divided by revenue.d. Good Months – where the profit after tax was greater than the mean for the year.e. Bad Months – where the profit after tax was less than the mean for the year.f. The best month – where the profit after tax was max for the year.g. The worst month – where the profit after tax was min for the year. <p>Note:</p> <ul style="list-style-type: none">a. All Results need to be presented as vectorsb. Results for Dollar values need to be calculated with \$0.01 precision, but need to be presented in Units of \$1000 (i.e 1k) with no decimal pointsc. Results for the profit margin ratio need to be presented in units of % with no decimal point.d. It is okay for tax to be negative for any given month (deferred tax asset)e. Generate CSV file for the data. <p>Suggested Reading – Text Book 1 – Chapter 4 (Vectors, Combining Matrices)</p>		
3	<p>Develop a program to create two 3 X 3 matrices A and B and perform the following operations a) Transpose of the matrix b) addition c) subtraction d) multiplication</p> <p>Suggested Reading – Text Book 1 – Chapter 4 (Matrices and Arrays – Array Arithmetic)</p>		

Template for Practical Course and if AEC is a practical Course Annexure-V

4	Develop a program to find the factorial of given number using recursive function calls. Suggested Reading – Reference Book 1 – Chapter 5 (5.5 – Recursive Programming) Text Book 1 – Chapter 8 (Flow Control and Loops – If and Else, Vectorized If, while loops, for loops), Chapter 6 (Creating and Calling Functions, Passing Functions to and from other functions)																				
5	Develop an R Program using functions to find all the prime numbers up to a specified number by the method of Sieve of Eratosthenes. Suggested Reading – Reference Book 1 - Chapter 5 (5.5 – Recursive Programming) Text Book 1 – Chapter 8 (Flow Control and Loops – If and Else, Vectorized If, while loops, for loops), Chapter 6 (Creating and Calling Functions, Passing Functions to and from other functions)																				
6	The built-in data set mammals contain data on body weight versus brain weight. Develop R commands to: a) Find the Pearson and Spearman correlation coefficients. Are they similar? b) Plot the data using the plot command. c) Plot the logarithm (log) of each variable and see if that makes a difference. Suggested Reading – Text Book 1 –Chapter 12 – (Built-in Datasets) Chapter 14 – (Scatterplots) Reference Book 2 – 13.2.5 (Covariance and Correlation)																				
7	Develop R program to create a Data Frame with following details and do the following operations. <table><tr><th>itemCode</th><th>itemCategory</th><th>itemPrice</th></tr><tr><td>1001</td><td>Electronics</td><td>700</td></tr><tr><td>1002</td><td>Desktop Supplies</td><td>300</td></tr><tr><td>1003</td><td>Office Supplies</td><td>350</td></tr><tr><td>1004</td><td>USB</td><td>400</td></tr><tr><td>1005</td><td>CD Drive</td><td>800</td></tr></table> a) Subset the Data frame and display the details of only those items whose price is greater than or equal to 350. b) Subset the Data frame and display only the items where the category is either “Office Supplies” or “Desktop Supplies” c) Create another Data Frame called “item-details” with three different fields itemCode, ItemQtyonHand and ItemReorderLvl and merge the two frames Suggested Reading –Textbook 1: Chapter 5 (Lists and Data Frames)			itemCode	itemCategory	itemPrice	1001	Electronics	700	1002	Desktop Supplies	300	1003	Office Supplies	350	1004	USB	400	1005	CD Drive	800
itemCode	itemCategory	itemPrice																			
1001	Electronics	700																			
1002	Desktop Supplies	300																			
1003	Office Supplies	350																			
1004	USB	400																			
1005	CD Drive	800																			
8	Let us use the built-in dataset air quality which has Daily air quality measurements in New York, May to September 1973. Develop R program to generate histogram by using appropriate arguments for the following statements. a) Assigning names, using the air quality data set. b) Change colors of the Histogram c) Remove Axis and Add labels to Histogram d) Change Axis limits of a Histogram e) Add Density curve to the histogram Suggested Reading –Reference Book 2 – Chapter 7 (7.4 – The ggplot2 Package), Chapter 24 (Smoothing and Shading)																				

Template for Practical Course and if AEC is a practical Course Annexure-V

9	<p>Design a data frame in R for storing about 20 employee details. Create a CSV file named "input.csv" that defines all the required information about the employee such as id, name, salary, start_date, dept. Import into R and do the following analysis.</p> <ol style="list-style-type: none"> Find the total number rows & columns Find the maximum salary Retrieve the details of the employee with maximum salary Retrieve all the employees working in the IT Department. Retrieve the employees in the IT Department whose salary is greater than 20000 and write these
	<p>details into another file "output.csv"</p> <p>Suggested Reading – Text Book 1 – Chapter 12(CSV and Tab Delimited Files)</p>
10	<p>Using the built in dataset mtcars which is a popular dataset consisting of the design and fuel consumption patterns of 32 different automobiles. The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973-74 models). Format A data frame with 32 observations on 11 variables : [1] mpg Miles/(US) gallon, [2] cyl Number of cylinders [3] disp Displacement (cu.in.), [4] hp Gross horsepower [5] drat Rear axle ratio,[6] wt Weight (lb/1000) [7] qsec 1/4 mile time, [8] vs V/S, [9] am Transmission (0 = automatic, 1 = manual), [10] gear Number of forward gears, [11] carb Number of carburetors</p> <p>Develop R program, to solve the following:</p> <ol style="list-style-type: none"> What is the total number of observations and variables in the dataset? Find the car with the largest hp and the least hp using suitable functions Plot histogram / density for each variable and determine whether continuous variables are normally distributed or not. If not, what is their skewness? What is the average difference of gross horse power(hp) between automobiles with 3 and 4 number of cylinders(cyl)? Also determine the difference in their standard deviations. Which pair of variables has the highest Pearson correlation? <p>References (Web links):</p> <ol style="list-style-type: none"> https://cran.r-project.org/web/packages/explore/vignettes/explore_mtcars.html https://www.w3schools.com/r/r_stat_data_set.asp https://rpubs.com/BillB/217355
11	<p>Demonstrate the progression of salary with years of experience using a suitable data set (You can create your own dataset). Plot the graph visualizing the best fit line on the plot of the given data points. Plot a curve of Actual Values vs. Predicted values to show their correlation and performance of the model. Interpret the meaning of the slope and y-intercept of the line with respect to the given data. Implement using lm function. Save the graphs and coefficients in files. Attach the predicted values of salaries as a new column to the original data set and save the data as a new CSV file.</p> <p>Suggested Reading – Reference Book 2 – Chapter 20 (General Concepts, Statistical Inference, Prediction)</p>
<p>Course outcomes (Course Skill Set):</p> <p>At the end of the course the student will be able to:</p> <ul style="list-style-type: none"> Explain the fundamental syntax of R data types, expressions and the usage of the R-Studio IDE • Develop a program in R with programming constructs: conditionals, looping and functions. Apply the list and data frame structure of the R programming language. Use visualization packages and file handlers for data analysis.. 	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation

rubrics shall be decided jointly by examiners.

14.09.2023

- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)

Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.

The minimum duration of SEE is 02 hours

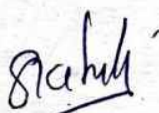
Suggested Learning Resources:

Book:

1. Cotton, R. (2013). Learning R: A Step by Step Function Guide to Data Analysis. 1st ed. O'Reilly Media Inc.

References:

1. Jones, O., Maillardet, R. and Robinson, A. (2014). Introduction to Scientific Programming and Simulation Using R. Chapman & Hall/CRC, The R Series.
2. Davies, T.M. (2016) The Book of R: A First Course in Programming and Statistics. No Starch Press.


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar, MOOBBIDRI - 574 225

Data Visualization with Python		Semester	III
Course Code	BCS358D	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0: 0: 2: 0	SEE Marks	50
Credits	01	Exam Hours	100
Examination type (SEE)	Practical		
Course objectives:			
<ul style="list-style-type: none">• CLO 1. Demonstrate the use of IDLE or PyCharm IDE to create Python Applications• CLO 2. Using Python programming language to develop programs for solving real-world problems• CLO 3. Implementation of Matplotlib for drawing different Plots• CLO 4. Demonstrate working with Seaborn, Bokeh.• CLO 5. Working with Plotly for 3D, Time Series and Maps.			
	Experiments		
Sl. No.	PART A – List of problems for which student should develop program and execute in the Laboratory		
1	<p>a) Write a python program to find the best of two test average marks out of three test's marks accepted from the user.</p> <p>b) Develop a Python program to check whether a given number is palindrome or not and also count the number of occurrences of each digit in the input number.</p> <p>Datatypes: https://www.youtube.com/watch?v=gCCVsvgR2KU Operators: https://www.youtube.com/watch?v=v5MR5JnKcZI Flow Control: https://www.youtube.com/watch?v=PqFKRqpHrjw For loop: https://www.youtube.com/watch?v=0ZvaDa8eT5s While loop: https://www.youtube.com/watch?v=HZARImviDxg Exceptions: https://www.youtube.com/watch?v=6SPDvPK38tw</p>		
2	<p>a) Defined as a function F as $F_n = F_{n-1} + F_{n-2}$. Write a Python program which accepts a value for N (where $N > 0$) as input and pass this value to the function. Display suitable error message if the condition for input value is not followed.</p> <p>b) Develop a python program to convert binary to decimal, octal to hexadecimal using functions.</p> <p>Functions: https://www.youtube.com/watch?v=BVfCWuca9nw Arguments: https://www.youtube.com/watch?v=ijXMGpoMkhQ Return value: https://www.youtube.com/watch?v=nuNXiEDnM44</p>		

8	<p>Write a Python program to explain working with bokeh line graph using Annotations and Legends.</p> <p>a) Write a Python program for plotting different types of plots using Bokeh.</p> <p>https://www.youtube.com/watch?v=HDvxYoRadcA</p>
9	<p>Write a Python program to draw 3D Plots using Plotly Libraries.</p> <p>https://www.youtube.com/watch?v=cCck7hCanpw&list=PLE50-dh6JzC4onXqkv9H3HtPbBVA8M94&index=4</p>

10	<p>a) Write a Python program to draw Time Series using Plotly Libraries.</p> <p>b) Write a Python program for creating Maps using Plotly Libraries.</p> <p>https://www.youtube.com/watch?v=xnJ2TNrGYik&list=PLE50-dh6JzC4onXqkv9H3HtPbBVA8M94&index=5</p> <p>https://www.youtube.com/watch?v=D35m2CdMhVs&list=PLE50-dh6JzC4onXqkv9H3HtPbBVA8M94&index=6</p>
----	---

Python (Full Course): https://www.youtube.com/watch?v=_uQrJ0TkZlc

Pedagogy For the above experiments the following pedagogy can be considered. Problem based learning, Active learning, MOOC, Chalk &Talk

Course outcomes (Course Skill Set):

At the end of the course the student will be able to:

- CO 1. Demonstrate the use of IDLE or PyCharm IDE to create Python Applications
- CO 2. Use Python programming constructs to develop programs for solving real-world problems
- CO 3. Use Matplotlib for drawing different Plots
- CO 4.

Demonstrate working with Seaborn, Bokeh for visualization.

- CO 5. Use Plotly for drawing Time Series and Maps.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks). The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.


The minimum duration of SEE is 02 hours

- Weightage of marks for PART A is 80% and for PART B is 20%. General rubrics suggested to be followed for part A and part B.
- Change of experiment is allowed only once and Marks allotted to the procedure part to be made zero (Not allowed for Part B).
- The duration of SEE is 03 hours

Rubrics suggested in Annexure-II of Regulation book

Textbooks:

1. Al Sweigart, "Automate the Boring Stuff with Python", 1st Edition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at <https://automatetheboringstuff.com/>)
2. Reema Thareja "Python Programming Using Problem Solving Approach" Oxford University Press.
3. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at <http://greenteapress.com/thinkpython2/thinkpython2.pdf>)
4. Jake VanderPlas "Python Data Science Handbook" 1st Edition, O'REILLY.


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miar. MOODBIDRI - 574 225

Analysis & Design of Algorithms		Semester	4
Course Code	BCS401	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
Course objectives: <ul style="list-style-type: none">• To learn the methods for analyzing algorithms and evaluating their performance.• To demonstrate the efficiency of algorithms using asymptotic notations.• To solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound.• To learn the concepts of P and NP complexity classes.			
Teaching-Learning Process (General Instructions) <p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none">1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.2. Utilize video/animation films to illustrate the functioning of various concepts.3. Promote collaborative learning (Group Learning) in the class.4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.6. Introduce topics through multiple representations.7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions.8. Discuss the real-world applications of every concept to enhance students' comprehension.			
Module-1			
INTRODUCTION: What is an Algorithm?, Fundamentals of Algorithmic Problem Solving. FUNDAMENTALS OF THE ANALYSIS OF ALGORITHM EFFICIENCY: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms. BRUTE FORCE APPROACHES: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching. Chapter 1 (Sections 1.1,1.2), Chapter 2(Sections 2.1,2.2,2.3,2.4), Chapter 3(Section 3.1,3.2)			
Module-2			

BRUTE FORCE APPROACHES (contd.): Exhaustive Search (Travelling Salesman problem and Knapsack Problem).

DECREASE-AND-CONQUER: Insertion Sort, Topological Sorting.

DIVIDE AND CONQUER: Merge Sort, Quick Sort, Binary Tree Traversals, Multiplication of Large Integers and Strassen's Matrix Multiplication.

Chapter 3(Section 3.4), Chapter 4 (Sections 4.1,4.2), Chapter 5 (Section 5.1,5.2,5.3, 5.4)

Module-3

TRANSFORM-AND-CONQUER: Balanced Search Trees, Heaps and Heapsort.

SPACE-TIME TRADEOFFS: Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm.

Chapter 6 (Sections 6.3,6.4), Chapter 7 (Sections 7.1,7.2)

Module-4

DYNAMIC PROGRAMMING: Three basic examples, The Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.

THE GREEDY METHOD: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes.

Chapter 8 (Sections 8.1,8.2,8.4), Chapter 9 (Sections 9.1,9.2,9.3,9.4)

Module-5

LIMITATIONS OF ALGORITHMIC POWER: Decision Trees, P, NP, and NP-Complete Problems.

COPING WITH LIMITATIONS OF ALGORITHMIC POWER: Backtracking (n-Queens problem, Subset-sum problem), Branch-and-Bound (Knapsack problem), Approximation algorithms for NP-Hard problems (Knapsack problem).

Chapter 11 (Section 11.2, 11.3), Chapter 12 (Sections 12.1,12.2,12.3)

Course outcome (Course Skill Set)

At the end of the course, the student will be able to:

1. Apply asymptotic notational method to analyze the performance of the algorithms in terms of time complexity.
2. Demonstrate divide & conquer approaches and decrease & conquer approaches to solve computational problems.
3. Make use of transform & conquer and dynamic programming design approaches to solve the given real world or complex computational problems.
4. Apply greedy and input enhancement methods to solve graph & string based computational problems.
5. Analyse various classes (P, NP and NP Complete) of problems
6. Illustrate backtracking, branch & bound and approximation methods.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally **reduced to 50 marks**

Suggested Learning Resources: Textbooks

1. Introduction to the Design and Analysis of Algorithms, By Anany Levitin, 3rd Edition (Indian), 2017, Pearson.

Reference books

1. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities Press.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
3. Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education)

Web links and Video Lectures (e-Resources):

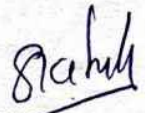
- Design and Analysis of Algorithms: <https://nptel.ac.in/courses/106/101/106101060/>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Promote real-world problem-solving and competitive problem solving through group discussions to engage students actively in the learning process.
- Encourage students to enhance their problem-solving skills by implementing algorithms and solutions through programming exercises, fostering practical application of theoretical concepts.

Assessment Methods -

1. Problem Solving Assignments (Hacker Rank/ Hacker Earth / Leadcode)
2. Gate Based Aptitude Test


 HOD's Signature
H.O.D.
 Dept. Of Information Science & Engineering
 Alva's Institute of Engg. & Technology
 Miiar, MOODBIDRI - 574 225

ADVANCED JAVA		Semester	IV
Course Code	BIS402	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	
Examination nature (SEE)	Theory/		
Note- Students who have not opted for Java course in earlier semester, student has to undergo a bridge course on basics of java before the commencement of 4th sem			
Course objectives:			
CLO 1. Understanding the fundamentals of collection framework			
CLO 2. Demonstrate the fundamental concepts of String operations and Swing applications			
CLO 3. Design and develop web applications using Java servlets and JSP			
CLO 4. Apply database interaction through Java database Connectivity			
Teaching-Learning Process (General Instructions)			
These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes.			
<ol style="list-style-type: none">1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.2. Promote collaborative learning (Group Learning) in the class.3. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.4. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.5. Introduce Topics in manifold representations.6. Demonstrate ways to solve the same problem and encourage the students to come up with their own creative solutions.7. Discuss application of every concept to solve the real world problems.			
MODULE-1			
The collections and Framework: Collections Overview, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Arrays,, The legacy Classes and Interfaces, Parting Thoughts on Collections.			
Text Book 1: Ch. 20			
MODULE-2			
String Handling : The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf(), Changing the Case of Characters Within a String, joining strings, Additional String Methods, StringBuffer , StringBuilder			
Text Book 1: Ch 18			
MODULE-3			

Introducing Swing: The Origin of Swing, Swing Is Built on AWT, Two Key Swing Features, The MVC Connection, Components and Containers, The Swing Packages, A Simple Swing Application, Event Handling, Painting in Swing,
Exploring Swing : JLabel and ImageIcon, JTextField, The Swing Buttons-JButton, JToggleButton, Check Boxes, Radio Buttons

Text Book 1: Ch 32 and Ch. 33

MODULE-4

Introducing servlets: Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Jakarta. Servlet Package; Reading Servlet Parameter; The Jakarta.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP); JSP tags, Variables and Objects, Methods, Control statements, Loops, Request String, Parsing other information, User sessions, Cookies, Session Objects.

Text Book 1: Ch 36 **Text Book 2:** Ch 11

MODULE-5

JDBC Objects: The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.

Text Book 2: Ch 06

PRACTICAL COMPONENT OF IPCC*(May cover all / major modules)*

Sl.N O	Experiments
1	Implement a java program to demonstrate creating an ArrayList, adding elements, removing elements, sorting elements of ArrayList. Also illustrate the use of toArray() method.
2	Develop a program to read random numbers between a given range that are multiples of 2 and 5, sort the numbers according to tens place using comparator.
3	Implement a java program to illustrate storing user defined classes in collection.
4	Implement a java program to illustrate the use of different types of string class constructors.
5	Implement a java program to illustrate the use of different types of character extraction, string comparison, string search and string modification methods.
6	Implement a java program to illustrate the use of different types of StringBuffer methods
7	Demonstrate a swing event handling application that creates 2 buttons Alpha and Beta and displays the text “Alpha pressed” when alpha button is clicked and “Beta pressed” when beta button is clicked.
8	A program to display greeting message on the browser “Hello UserName”, “How Are You?”, accept username from the client using servlet.
9	A servlet program to display the name, USN, and total marks by accepting student detail
10	A Java program to create and read the cookie for the given cookie name as “EMPID” and its value as “AN2356”.
11	Write a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries(For example update, delete, search etc...).
12	A program to design the Login page and validating the USER_ID and PASSWORD using JSP and DataBase.
Course outcomes (Course Skill Set): At the end of the course, the student will be able to: CO 1. Apply appropriate collection class/interface to solve the given problem CO 2. Demonstrate the concepts of String operations in Java CO 3. Apply the concepts of Swings to build Java applications CO 4. Develop web based applications using Java servlets and JSP CO 5. Use JDBC to build database applications	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks**).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC. **CIE for the practical component of the IPCC**

- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test (**duration 02/03 hours**) after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks**.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources: Books

1. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education, 2004.
3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.


Web links and Video Lectures (e-Resources):

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. <https://nptel.ac.in/courses/106/105/106105225/>
3. <https://youtu.be/qGMxs-PbFPk>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Demonstration of simple projects on database connectivity

Group assignment for J2EE projects


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miar. MOODBIDRI - 574 225

DATABASE MANAGEMENT SYSTEM		Semester	4
Course Code	BCS403	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	
Examination nature (SEE)	Theory		
Course objectives: <ul style="list-style-type: none">● To Provide a strong foundation in database concepts, technology, and practice.● To Practice SQL programming through a variety of database problems.● To Understand the relational database design principles.● To Demonstrate the use of concurrency and transactions in database.● To Design and build database applications for real world problems.● To become familiar with database storage structures and access techniques.			
Teaching-Learning Process <p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none">1. Lecturer method (L) needs not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.2. Use of Video/Animation to explain functioning of various concepts.3. Encourage collaborative (Group Learning) Learning in the class.4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.6. Introduce Topics in manifold representations.7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding9. Use any of these methods: Chalk and board, Active Learning, Case Studies			
MODULE-1			No. of Hours: 8
Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications.			
Overview of Database Languages and Architectures: Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment.			
Conceptual Data Modelling using Entities and Relationships: Entity types, Entity sets and structural constraints, Weak entity types, ER diagrams,Specialization and Generalization.			
Textbook 1: Ch 1.1 to 1.8, 2.1 to 2.6, 3.1 to 3.10			
RBT: L1, L2, L3			
MODULE-2			No. of Hours: 8

Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations.
Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra.
Mapping Conceptual Design into a Logical Design: Relational Database Design using ER-to-Relational mapping.

Textbook 1: Ch 5.1 to 5.3, Ch 8.1 to 8.5; Ch 9.1 to 9.2 **Textbook 2:** 3.5 **RBT:** L1, L2, L3

MODULE-3**No. of Hours:8**

Normalization: Database Design Theory – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

SQL: SQL data definition and data types, Schema change statements in SQL, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL

Textbook 1: Ch 14.1 to 14.7, Ch 6.1 to 6.5

RBT: L1, L2, L3

MODULE-4**No. of Hours:8**

SQL: Advanced Queries: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL.

Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL.

Textbook 1: Ch 7.1 to 7.3, Ch 20.1 to 20.6

RBT: L1, L2, L3

MODULE-5**No. of Hours:08**

Concurrency Control in Databases: Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking.

NOSQL Databases and Big Data Storage Systems: Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j

Textbook 1:Chapter 21.1 to 21.5, Chapter 24.1 to 24.6 **RBT:**

L1, L2, L3

PRACTICAL COMPONENT OF IPCC *(May cover all / major modules)*

Sl.NO	Experiments
-------	-------------

1	<p>Create a table called Employee & execute the following. Employee(EMPNO,ENAME,JOB, MANAGER_NO, SAL, COMMISSION)</p> <ol style="list-style-type: none"> 1. Create a user and grant all permissions to the user. 2. Insert the any three records in the employee table contains attributes EMPNO,ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result. 3. Add primary key constraint and not null constraint to the employee table. 4. Insert null values to the employee table and verify the result.
2	<p>Create a table called Employee that contain attributes EMPNO,ENAME,JOB, MGR,SAL & execute the following.</p> <ol style="list-style-type: none"> 1. Add a column commission with domain to the Employee table. 2. Insert any five records into the table. 3. Update the column details of job 4. Rename the column of Employee table using alter command. 5. Delete the employee whose Empno is 105.
3	<p>Queries using aggregate functions(COUNT,AVG,MIN,MAX,SUM),Group by,Orderby. Employee(E_id, E_name, Age, Salary)</p> <ol style="list-style-type: none"> 1. Create Employee table containing all Records E_id, E_name, Age, Salary. 2. Count number of employee names from employee table 3. Find the Maximum age from employee table. 4. Find the Minimum age from employee table. 5. Find salaries of employee in Ascending Order. 6. Find grouped salaries of employees.
4	<p>Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary. CUSTOMERS(ID,NAME,AGE,ADDRESS,SALARY)</p>
5	<p>Create cursor for Employee table & extract the values from the table. Declare the variables ,Open the cursor & extract the values from the cursor. Close the cursor. Employee(E_id, E_name, Age, Salary)</p>
6	<p>Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.</p>
7	<p>Install an Open Source NoSQL Data base MongoDB & perform basic CRUD(Create, Read, Update & Delete) operations. Execute MongoDB basic Queries using CRUD operations.</p>

Course outcomes (Course Skill Set):

At the end of the course, the student will be able to:

- Describe the basic elements of a relational database management system ● Design entity relationship for the given scenario.
- Apply various Structured Query Language (SQL) statements for database manipulation.
- Analyse various normalization forms for the given application.
- Develop database applications for the given real world problem.
- Understand the concepts related to NoSQL databases.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum

passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks**).
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC. **CIE for the practical component of the IPCC**
- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test (**duration 02/03 hours**) after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks**.
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 subquestions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

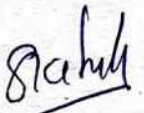
Suggested Learning Resources:

Text Books:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning Mini Project:

- Project Based Learning


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar. MOODBIDRI - 574 225

Analysis & Design of Algorithms Lab		Semester	4
Course Code	BCSL404	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	2
Examination type (SEE)	Practical		
Course objectives: <ul style="list-style-type: none">• To design and implement various algorithms in C/C++ programming using suitable development tools to address different computational challenges.• To apply diverse design strategies for effective problem-solving.• To Measure and compare the performance of different algorithms to determine their efficiency and suitability for specific tasks.			
Sl.No	Experiments		
1	Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm.		
2	Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.		
3	a. Design and implement C/C++ Program to solve All-Pairs Shortest Paths problem using Floyd's algorithm. b. Design and implement C/C++ Program to find the transitive closure using Warshal's algorithm.		
4	Design and implement C/C++ Program to find shortest paths from a given vertex in a weighted connected graph to other vertices using Dijkstra's algorithm.		
5	Design and implement C/C++ Program to obtain the Topological ordering of vertices in a given digraph.		
6	Design and implement C/C++ Program to solve 0/1 Knapsack problem using Dynamic Programming method.		
7	Design and implement C/C++ Program to solve discrete Knapsack and continuous Knapsack problems using greedy approximation method.		
8	Design and implement C/C++ Program to find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d .		
9	Design and implement C/C++ Program to sort a given set of n integer elements using Selection Sort method and compute its time complexity. Run the program for varied values of $n > 5000$ and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
10	Design and implement C/C++ Program to sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of $n > 5000$ and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
11	Design and implement C/C++ Program to sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of $n > 5000$, and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
12	Design and implement C/C++ Program for N Queen's problem using Backtracking.		

Course outcomes (Course Skill Set):

At the end of the course the student will be able to:

1. Develop programs to solve computational problems using suitable algorithm design strategy.
2. Compare algorithm design strategies by developing equivalent programs and observing running times for analysis (Empirical).
3. Make use of suitable integrated development tools to develop programs
4. Choose appropriate algorithm design techniques to develop solution to the computational and complex problems.
5. Demonstrate and present the development of program, its execution and running time(s) and record the results/inferences.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)

Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.


The minimum duration of SEE is 02 hours

Suggested Learning Resources:

- Virtual Labs (CSE): <http://cse01-iiith.vlabs.ac.in/>

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.


HOD's Signature
H.O.D.
 Dept. Of Information Science & Engineering
 Alva's Institute of Engg. & Technology
 Miiar, MOODBIDRI - 574 225

DISCRETE MATHEMATICAL STRUCTURES		Semester	IV
Course Code	BCS405A	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
Course objectives: <ol style="list-style-type: none">1. To help students to understand discrete and continuous mathematical structures.2. To impart basics of relations and functions.3. To facilitate students in applying principles of Recurrence Relations to find the generating functions and solve the Recurrence relations.4. To have the knowledge of groups and their properties to understand the importance of algebraic properties relative to various number systems.			
Teaching-Learning Process Pedagogy (General Instructions): These are sample Strategies, teachers can use to accelerate the attainment of the various course outcomes. <ol style="list-style-type: none">1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills.2. State the need for Mathematics with Engineering Studies and Provide real-life examples.3. Support and guide the students for self-study.4. You will assign homework, grading assignments and quizzes, and documenting students' progress.5. Encourage the students to group learning to improve their creative and analytical skills.6. Show short related video lectures in the following ways:<ul style="list-style-type: none">• As an introduction to new topics (pre-lecture activity).• As a revision of topics (post-lecture activity).• As additional examples (post-lecture activity).• As an additional material of challenging topics (pre-and post-lecture activity).• As a model solution for some exercises (post-lecture activity).			
Module-1: Fundamentals of Logic Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems. (8 hours) (RBT Levels: L1, L2 and L3)			
Module-2: Properties of the Integers Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions. Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition. (8 Hours) (RBT Levels: L1, L2 and L3)			
Module-3: Relations and Functions			

<p>Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeonhole Principle, Function Composition and Inverse Functions.</p> <p>Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions. (8 hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>
<p align="center">Module-4: The Principle of Inclusion and Exclusion</p>
<p>The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.</p> <p>Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients. (8 Hours) (RBT Levels: L1, L2 and L3)</p>
<p align="center">Module-5: Introduction to Groups Theory</p>
<p>Definitions and Examples of Particular Groups Klein 4-group, Additive group of Integers modulo n, Multiplicative group of Integers modulo p and permutation groups, Properties of groups, Subgroups, cyclic groups, Cosets, Lagrange's Theorem. (8 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>
<p>Course outcome (Course Skill Set)</p> <p>At the end of the course, the student will be able to:</p> <ol style="list-style-type: none"> 1. Apply concepts of logical reasoning and mathematical proof techniques in proving theorems and statements. 2. Demonstrate the application of discrete structures in different fields of computer science. 3. Apply the basic concepts of relations, functions and partially ordered sets for computer representations. 4. Solve problems involving recurrence relations and generating functions. 5. Illustrate the fundamental principles of Algebraic structures with the problems related to computer science & engineering.
<p>Assessment Details (both CIE and SEE)</p> <p>The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE, the minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.</p>

Continuous Internal Evaluation:

- There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component.
- Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks
- Any two assignment methods mentioned in the 220B2.4, if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks) The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

The Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by the University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.

Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year) Text Books:

1. **Ralph P. Grimaldi, B V Ramana: "Discrete Mathematical Structures an Applied Introduction"**, 5th Edition, Pearson Education, 2004.
2. **Ralph P. Grimaldi: "Discrete and Combinatorial Mathematics"**, 5th Edition, Pearson Education. 2004.

Reference Books:


1. **Basavaraj S Anami and Venakanna S Madalli: "Discrete Mathematics – A Concept-based approach"**, Universities Press, 2016
2. **Kenneth H. Rosen: "Discrete Mathematics and its Applications"**, 6th Edition, McGraw Hill, 2007.
3. **Jayant Ganguly: "A Treatise on Discrete Mathematical Structures"**, Sanguine-Pearson, 2010.
4. **D.S. Malik and M.K. Sen: "Discrete Mathematical Structures Theory and Applications"**, Latest Edition, Thomson, 2004.
5. **Thomas Koshy: "Discrete Mathematics with Applications"**, Elsevier, 2005, Reprint 2008.

Web links and Video Lectures (e-Resources):

- <http://nptel.ac.in/courses.php?disciplineID=111>
- [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
- <http://academicearth.org/> • VTU e-Shikshana Program
- VTU EDUSAT Program.
- <http://www.themathpage.com/>
- <http://www.abstractmath.org/>
- <http://www.ocw.mit.edu/courses/mathematics/>

Activity-Based Learning (Suggested Activities in Class)/Practical-Based Learning

- Quizzes
- Assignments
- Seminar


 HOD's Signature
H.O.D.
 Dept. Of Information Science & Engineering
 Alva's Institute of Engg. & Technology
 Miiar, MOODBIDRI - 574 225

Green IT and Sustainability		Semester	4
Course Code	BCS456A	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	14	Total Marks	100
Credits	01	Exam Hours	01
Examination type (SEE)	Theory(MCQ)		
Course objectives: <ul style="list-style-type: none">Understand challenges for Green ICT and the environmental impact.Learn different aspects of ICT metrics and Sustainable Cloud Computing.Explore effects of software design on the sustainability.			
Teaching-Learning Process (General Instructions) <p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none">Lecturer method (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.Use of Video/Animation to explain functioning of various concepts.Encourage collaborative (Group Learning) Learning in the class.Ask at least three HOT (Higher order Thinking) questions in the class, which promotes Critical thinking.Adopt Case study Based Learning (CBL), which fosters students' analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.			
Module-1			
Green ICT -History, Agenda, and Challenges Ahead: Introduction, Industrial Revolution, The Emergence of Information and Communication Technologies, The Agenda and Challenges Ahead.			
Module-2			
Emerging Technologies and Their Environmental Impact: Introduction, Number of Connected Devices , Increased , Functionality, Increased Number of Separate Functions , Increased Demand for Speed and Reliability , Obsolescence—The Problem of Backward Compatibility, The Other Side of the Balance Sheet, Videoconference as an Alternative to Business Travel, Dematerialization of Product Chain, Travel Advice/Road Traffic Control, Intelligent Energy Metering , Building Management Systems, Saving IT			
Resources.	Module-3		
Measurements and Sustainability: Introduction, ICT Technical Measures, Ecological Measures and Ethical Consideration, Systems Engineering for Designing Sustainable ICT-Based Architectures.			
Module-4			
Sustainable Cloud Computing: Introduction, Challenges in the Use of Cloud Computing As Green Technology, Cloud Computing and Sustainability, Sustainable Applications of Cloud Computing, Technologies Associated With Sustainable Cloud Computing, Future Prospects of Sustainable Cloud Computing, Reflections on Sustainable Cloud Computing Applications.			
Module-5			

Sustainable Software Design: Overview and Scope, Evaluating Sustainability Effects , Sustainability and the Product Life Cycle , Direct Effects: Sustainability During Use, Runtime Energy Consumption Basics , Analyzing the Energy Consumption of an Application , Energy Consumption Reduction Using Physical Properties of Semiconductors, Optimizing the Energy Consumption of an Application: Compiler Techniques, Optimizing the Energy Consumption of an Application: Runtime Approaches.

Course outcome (Course Skill Set)

At the end of the course the student will be able to:

1. Classify the challenges for Green ICT
2. Relate the environmental impact due to emerging technologies.
3. Demonstrate different aspects of ICT metrics.
4. Compare the various parameters related to Sustainable Cloud Computing.
5. Interpret the effects of software design on the sustainability.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous internal Examination (CIE)

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examinations (SEE)

SEE paper shall be set for 50 questions, each of the 01 marks. The pattern of the question paper is MCQ (multiple choice questions). The time allotted for SEE is **01 hour**. The student has to secure a minimum of 35% of the maximum marks meant for SEE.

Suggested Learning Resources: Books


1. Green Information Technology – A Sustainable Approach, Mohammad Dastbaz Colin Pattinson, Babak Akhgar, Elsevier, 2015 Inc.
2. San Murugesan; G. R. Gangadharan, Harnessing Green IT: Principles and Practices, Wiley-IEEE Press

Web links and Video Lectures (e-Resources):

- https://www.youtube.com/watch?v=kvn_-mJ2tSo
- <https://www.youtube.com/watch?v=kxngsYn5N3Y>
- <https://www.youtube.com/watch?v=EgdFi3sCgzU>
- <https://www.brightest.io/sustainability-measurement>
- <https://www.youtube.com/watch?v=S2m49Op25Zw>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Literature survey/review


HOD's Signature
H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Miiar. MOODBIDRI - 574 225