Subject Title	**Mathematics for Computer Science**		
Subject Code	**BCS301**	**IA Marks**	50
Number of Lecture Hrs /	03	**Exam Marks**	50
Total Number of Lecture Hrs	40	**Exam Hours**	03
CREDITS – 04			

FACULTY DETAILS:		
Name: Dr. S. L. Patil	Designation: Asst. Professor	Experience: 14
Name: Prof. S. A. Patil	Designation: Asst. Professor	Experience: 12
No. of times course taught:01	**Specialization:** Mathematics	

1.0 Prerequisite Subjects:

Sl. No	Branch	Semester	Subject
01	Computer Science and Engineering	I	Mathematics-I for CSE

2.0 Course Objectives

Course Learning Objectives:

- 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.
- To Provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypotheses.
- To Determine whether an input has a statistically significant effect on the system's response through ANOVA testing.

3.0 Course Outcomes

Having successfully completed this course, the student will be able to

Course Code	Course Outcome	RBTL	POs
C201.1	Explain the basic concepts of probability, random variables, probability distribution	L1,L2,L3	1,2,3,12
C201.2	Apply suitable probability distribution models for the given scenario.	L1,L2,L3	1,2,3,12
C201.3	Apply the notion of a discrete-time Markov chain and n-step transition	L1,L2,L3	1,2,3,12
C201.4	Use statistical methodology and tools in the engineering problem-solving process.	L1,L2,L3	1,2,3,12
C201.5	Apply the ANOVA test related to engineering problems.	L1,L2,L3	1,2,3,12
Total Hours of instruction		**40**	

4.0 Course Content

Module-1: 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)**

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)**

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)**

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 distributions a test of goodness of fit. F-Distribution. **(12 hours)**

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)**

5.0 Relevance to future subjects

Sl. No.	Semester	Subject	Topics
01	Common to all	Common to all engineering Subjects	Signal and Analysis, Field Theory, Thermodynamics, Fluid Dynamics etc.

6.0 Relevance to Real World

Sl.No	Real World Mapping
01	Probability is widely used in all sectors in daily life like sports, weather reports, blood samples, predicting the sex of the baby in the womb, congenital disabilities, statics, and many. In this topic, we will learn in detail about probability.
02	stochastic process it is widely used in statistical physics, such as weather forecast astrophysics strategy decision population theory reliability of safety science and technology of economic mathematics
03	Sampling is very often used in our daily life. For example, while purchasing fruits from a shop, we usually examine a few to assess the quality. A doctor examines a few drops of blood as a sample and draws a conclusion about the blood constitution of the whole body.
04	The ANOVA test can be used in the business world. If a company wants to test the effectiveness of five different marketing strategies, they will use the ANOVA test.

7.0 Gap Analysis and Mitigation

Sl. No	Delivery Type	Details
01	Tutorial	Topic: Differential equations

8.0 Books Used and Recommended to Students

Text Books

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

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 & 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 & its Applications", Vol.1, Wiley, 3rd Ed., 1968.
13.N.P. Bali and Manish Goyal, A Textbook of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
12.Veerarajan T, Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010

9.0 Relevant Websites (Reputed Universities and Others) for Notes/Animation/Videos Recommended

Website and Internet Contents References

Web links and Video Lectures:

1. http://nptel.ac.in/courses.php?disciplineID=111
2. http://www.class-central.com/subject/math(MOOCs)
3. http://academicearth.org/
4. VTU Edusat Programme
5. VTU e-Shikshana Program
6. http://www.bookstreet.in.

10.0 Magazines/Journals Used and Recommended to Students

Sl.No	Magazines/Journals	website
1	+ Plus Magazine	https://plus.maths.org/issue44.
2	Mathematics Magazine	www.mathematicsmagazine.com

11.0 Examination Note

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 of50) 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:

1.For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

2.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.

3. 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.

4.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. **Marks scored shall be proportionally reduced to 50 marks**

12.0 Course Delivery Plan

Module No.	Lecture No.	Content of Lecturer	% of Portion
1	1	Review of basic probability theory	20
	2	Random variables (discrete and continuous)	
	3	Probability mass/density functions	
	4	Mathematical expectation, mean and variance..	
	5	Binomial distribution	
	6	Poisson distribution.	
	7	derivations for mean and standard deviation for Binomial and Poisson distributions	
	8	Problems	
	9	Normal distributions.	
	10	Problems	
	11	Exponential distribution.	
	12	Problems	
2	13	Joint Probability distribution for two discrete random variables	20
	14	Expectation	
	15	Covariance and Correlation	
	16	Problems	
	17	Introduction to Stochastic Process	
	18	Probability Vectors	
	19	Stochastic matrices	
	20	Regular stochastic matrices	
	21	Markov chains	
	22	Higher transition probabilities	
	23	Stationary distribution of Regular Markov chains	
	24	Absorbing states	
3	25	Introduction	20
	26	Sampling Distribution	
	27	Standard Error	
	28	Testing of Hypothesis	
	29	Levels of Significance	
	30	Test of Significances	
	31	Confidence Limits	
	32	Simple Sampling of Attributes	
	33	Test Of Significance for Large Samples	
	34	Problems	
	35	Comparison of Large Samples	
	36	Problems	
4	37	Sampling variables	
	38	central limit theorem	
	39	confidences limit for unknown mean	
	40	Problems	

	41	Test of Significance for means of two small samples	20
	42	Problems	
	43	students 't' distribution	
	44	Problems	
	45	Chi-square distribution as a test of goodness of fit	
	46	Problems	
	47	F-Distribution	
	48	Problems	
5	49	Principles of experimentation in design	20
	50	Problems	
	51	Analysis of completely randomized design	
	52	Problems	
	53	randomized block design	
	54	The ANOVA Technique	
	55	Basic Principle of ANOVA	
	56	One-way ANOVA	
	57	Two-way ANOVA	
	58	Problems	
	59	Latin-square Design	
	60	Analysis of Co-Variance	

13.0 Assignments

Sl. No.	Title	Outcome expected	Allied study	Week No.	Individual / Group activity	Reference: book/website /Paper
1	Assignment 1: University Questions	Students study the Topics and write the Answers. Get practice to solve university questions.	Module 1,2&3 of the syllabus	6	Individual Activity.	Book 1, of the reference list. Website of the Reference list
2	Assignment 2: University Questions	Students study the Topics and write the Answers. Get practice to solve university questions.	Module 4, &5 of the syllabus	10	Individual Activity.	Book 1, 2 of the reference list. Website ofthe Reference list

14.0 QUESTION BANK

Module-1: Probability Distributions

1. Find the mean & variance of Binomial distribution.
2. The marks of 1000 students in an examination follows in a normal distribution with mean 70 & SD 5. Find the number of students whose marks will be i) less than 65, ii) more than 75 & iii) between 65 & 75.
3. The probability mass function of a variate X is

$X = x_i$	-2	-1	0	1	2	3
$p(x)$	0.1	K	0.2	2k	0.3	k

4. Find i) The value of K, ii) ≤ 0), iii) > 1) iv) $2 < x \leq 1$)

5. If 10% of the rivets produced by a machine are defective, find the probability that, out of 12 rivets chosen at random.
6. In a test of 2000 electric bulbs, it was found that the life of a bulb is a normal variable with average life of 2040 hours & standard deviation of 60 hours. Estimate the number of bulbs to burn for i) More than 2150 hours , ii) less than 1950 hours , Given that p[$0 \le z \le 1.83$] = 0.4664 &p[$0 \le z \le 1.33$] = 0.4082.
7. 2% of the fusion manufactured by a firm are found to be defective. Find the probability that a box containing 200 fuses contains i) no defective fuse , ii) 3 or more defective fuses.
8. In length of a telephone conversation is an exponential vitiate with mean 3 minutes. Find the probability that call i) ends in less than 3 minutes, ii) takes between 3 to 5 minutes.
9. Suppose that the student IQ scores form a normal distribution with average 100 & standard deviation 20. Find the percentage of students whose (i) score less than 80 (ii) score more than 120 (iii) score falls between 80 & 120 (G T P(1)=0.3413)
10. In a certain town the duration of a shower is exponentially distributed with mean 5 minutes what is the probability that a shower will least for i) 10 minutes or more, ii) less than 10 minutes, iii) between 10 min &12 min
11. The probability that a person aged 60 years will live up to 70 is 0.65. what is the probability that out of 10 persons aged 60 at least 7 of them will live up to 70.

Module-2:Joint probability distribution & Markov Chain

1. The joint probability distribution for two random variables X and Y is as given below.

X \ Y	-2	-1	4	5
1	0.1	0.2	0	0.3
2	0.2	0.1	0.1	0

Find the marginal distributions of X, Y. Also find the covariance of X and Y.

2. The Joint probability distribution of two random variables X and Y is as follows

X \ Y	-4	2	7
1	1/8	1/4	1/8
5	1/4	1/8	1/8

Determine (i) Marginal distribution of X & Y (ii) E(X) , E(Y) and E(XY) (iii) Cov (XY) (iv) $\rho(XY)$.

3. The Joint distribution of two random variables X and Y is as follows

X \ Y	-3	2	4
1	0.1	0.2	0.2
2	0.3	0.1	0.1

Compute (i) E(X) and E(Y), (ii) E(XY), (iii) Cov (XY), (iv) $\rho(XY)$

4. Find the fixed probability vector of the regular stochastic matrix. $3\begin{bmatrix} 1/3 & 1/3 \\ 0 & 1 \\ 1/2 & 0 \end{bmatrix}$

5. Find the fixed probability vector of the regular stochastic matrix. $6\begin{bmatrix} 1 & 0 \\ 1/2 & 1/3 \\ 2/3 & 1/3 \end{bmatrix}$

6. Define stochastic matrix. Find the unique fixed probability vector for the regular stochastic matrix

$$\begin{bmatrix} 0 & 1 & 0 \\ 1/2 & 0 & 1/2 \\ 1/2 & 1/4 & 1/4 \end{bmatrix}$$

7. Each year a man trades his car for a new car in 3 brands of the popular company Maruti Udyuog Limited. If he has a 'standard' he trades it for 'zen'. If he has a 'zen' he trades it for a 'Esteem'. If he has a 'Esteem' is just as likely to trade it for a new 'Esteem' or for a 'zen' or a 'standard'. In 1996 he bought his first car which was 'Esteem'. Find the probability that he has (i) 1999 Esteem (ii)1998 Standard (iii)1999 Zen

Module-3: Statistical Inference 1

1. Explain the following terms i) Null hypothesis , ii) Level of significance , iii) Type I & II errors , iv) Confidence limits.
2. A sample of 100 days is taken from meteorological records of certain districts & 10 of them are found to be fussy. Find the 99.73 % confidence interval of the % of fussy days in the distinct.
3. A die was thrown 9000 times & a throw of 5 or 6 was obtained 3240 times. On the assumption of random throwing, do the data abdicate that the die is biased?
4. A random sample of 100 records deaths in past year showed an average life span of 71.8 years. Assuming a population standard deviation of 8.9 years, does the data indicated that average life span today is greater than 70 years? Use a 0.05 level of significance.
5. In 324 throws of a six faced die, an odd number turned up 181 times. Is it reasonable to think that the die is an unbiased one?
6. In 324 throws of a six faced die, an odd number turned up 181 times. Is it reasonable to think that the die is an unbiased one? And define i) Confidence limits, ii) Level of significance, iii) Type I & II errors.
7. A coin was tossed 400 times & the head turned up 216 times. Test the hypotheses that the coin is in biased at 5% level significance.
8. A die was thrown 1200 times & the number 6 was obtained 236 times. Can the die be considered fair at level of significance?
9. A fair coin is tossed 4 times. Let X denotes the number of heads occurring and let Y denotes the longest string of heads occurring. Find the joint distribution function of X and Y.

Module-4: Statistical Inference 2

1. Four coins are tossed 100 times & the following results were obtained

No. of Heads	0	1	2	3	4
Frequencies	5	29	36	25	5

Fit a Binomial distribution for the data & test the goodness of fit given $\chi^2_{0.05} = 9.49$ for 4 d. f

2. Find the student's 't' for the following variable values in a sample of eight -4,-2,-2,0,2,2,3,3 taking the mean of the universe to be zero.
3. A certain stimulus administered to each of the 12 patients resulted in the following blood pressure 5,2,8,-1,3,0,6,-2,1,5,0,4, can it be calculated that stimulus will increase the blood pressure ? [$t_{0.05}$ for 11d.f= 2.201]
4. Ten individuals are chosen at random from a population and their heights in inches are found to be 45, 53, 62, 82, 45, 79, 80, 80,53, 50. Discuss the suggestion that the mean height of the population is 63 inches.

5. A die is thrown 264 times & the number appearing on the face follows the following frequency distribution.

X	1	2	3	4	5	6
F(x)	40	32	28	58	54	60

calculate the value of χ^2.

6. Fit a binomial distribution for the data

o. Heads	0	1	2	3	4
requency	122	60	15	2	1

and also test the goodness of fit given data $\chi^2_{0.05} = 7.815$ for 3 d.f.

Module-5: Design of Experiments & ANOVA

1. Three different kinds of food are tested on three groups of rats for 5 weeks. The objective is to check the difference in mean weight(in grams) of the rats per week. Apply one-way ANOVA using a 0.05 significance level to the following data:

Food I	Food II	Food III
8	4	11
12	5	8
19	4	7
8	6	13
6	9	7
11	7	9

2. Calculate the ANOVA coefficient for the following data:

Plant	**Number**	**Average span**	**s**
Hibiscus	5	12	2
Marigold	5	16	1
Rose	5	20	4

3. The following data show the number of worms quarantined from the GI areas of four groups of muskrats in a carbon tetrachloride anthelmintic study. Conduct a two-way ANOVA test.

I	**II**	**III**	**IV**
338	412	124	389
324	387	353	432
268	400	469	255
147	233	222	133
309	212	111	265

4. The three samples below have been obtained from Normal population with equal variance. Test the hypothesis at 5% level that population means are equal

8	7	12
10	5	9
7	10	13
14	9	12
11	9	14
50	40	60

5.To assess the significance of possible variation in performance in a certain test between the school of A city, a common test was given to a number of students taken at random from the 12th class of the 3 school concerned. The results given below: Make the ANOVA for the given data

A	B	C
2	3	4
4	5	6
6	7	8

Make the analysis of variance for the given data.

16.0 University Result

Examination	FCD (S+, S, A)	FC (B)	SC (C, D, E)	% Passing

Prepared by	Checked by		
Dr.S. L. Patil	Prof.S. A. Patil	HOD	Principal

SUBJECT TITLE	Digital Design and Computer Organization		
Subject Code	BCS302	**CIE Marks**	50
Number of Lecture Hrs / Week	3:0:2:0	**SEE Marks**	50
Total Number of Lecture Hrs	40 T + 20 P	**Total Marks**	100
CREDITS	04	**Exam Hours**	03
Examination nature(SEE)	Theory (DIV-A)		

FACULTY DETAILS:		
Name: Prof .N. K. Honnagoudar	**Designation:** Asst. Professor	**Experience: 21**
No. of times course taught:08	**Specialization:** Electronics	

1.0 Prerequisite Subjects:

Sl. No	Branch	Semester	Subject
01	Computer Science Engineering	III	Analog And Digital Electronics

2.0 Course Objectives

This course will enable students to

1. To demonstrate the functionalities of binary logic system
2. To explain the working of combinational and sequential logic system
3. To realize the basic structure of computer system
4. To illustrate the working of I/O operations and processing unit

3.0 Course Outcomes

Having successfully completed this course, the student will be able to draw and use modeling software's to generate

CO	Course Outcome	Cognitive Level	POs
C202.1	Apply the K–Map techniques to simplify various Boolean expressions.	L1, L2,L3	1,2,3,4,6,8,12
C202.2	Design different types of combinational and sequential circuits along with Verilog programs.	L1, L2,L3	1,2,3,4,6,8,12
C202.3	Describe the fundamentals of machine instructions, addressing modes and Processor performance.	L1, L2,L3	1,2,3,4,6,8,12
C202.4	Explain the approaches involved in achieving communication between processor and I/O devices.	L1, L2,L3	1,2,3,4,6,8,12
C202.5	Analyze internal Organization of Memory and Impact of cache/Pipelining on Processor Performance.	L1, L2,L3	1,2,3,4,6,8,12
Total Hours of instruction			**40**

4.0 Course Content

MODULE-I **8 Hours**

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) RBT:L1,L2,L3

MODULE-II **8 Hours**

Combinational Logic: Introduction, Combinational Circuits, Design Procedure, Binary Adder- Subtractor, Decoders, Encoders, and 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.) RBT:L1,L2,L3

MODULE-III **8 Hours**

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) RBT:L1,L2,L3.

MODULE-IV **8 Hours**

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) RBT:L1,L2,L3.

MODULE-V **8 Hours**

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)) RBT:L1,L2,L3

PRACTICAL COMPONENT OF IPCC

SI NO	Experiments Simulation packages preferred: Multisim, Modelsim, PSpice or any other relevant
01	Given a 4-variable logic expression, simplify it using appropriate technique and simulate the same using basic gates.
02	Design a 4 bit full adder and subtractor and simulate the same using basic gates.
03	Design Verilog HDL to implement simple circuits using structural, Data flow and Behavioral model.
04	Design Verilog HDL to implement Binary Adder-Subtractor – Half and Full Adder, Half and Full Subtractor.

05	Design Verilog HDL to implement Decimal adder.
06	Design Verilog program to implement Different types of multiplexer like 2:1, 4:1 and 8:1.
07	Design Verilog program to implement types of De-Multiplexer.
08	Design Verilog program for implementing various types of Flip-Flops such as SR, JK and D.

5.0 Relevance to future subjects

Sl. No	Semester	Subject	Topics
01	III & VII	Logic systems	Logic design and analysis

6.0 Relevance to Real World

Sl. No	Real World Mapping
01	Logic methods are used to solve engineering problems.
02	Combination logic circuits are used to design several application
03	Analog to digital and digital to analog application in various fields

7.0 Gap Analysis and Mitigation

Sl. No	Delivery Type	Details
01	Tutorial	Topic: K-Map Method

8.0 Books Used and Recommended to Students

Text Books

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

Reference Books

Additional Study material & e-Books

1. Analog Electronic Circuits: https://nptel.ac.in/courses/108/102/108102112/
2. Digital Electronic Circuits: https://nptel.ac.in/courses/108/105/108105132/
3. Analog Electronics Lab: http://vlabs.iitkgp.ac.in/be/

4. Digital Electronics Lab: http://vlabs.iitkgp.ac.in/dec

9.0 Relevant Websites (Reputed Universities and Others) for Notes/Animation/Videos Recommended

Website and Internet Contents References
1. www.iitg.ac.in/apvajpeyi/ph218/PH-218%20-%20Introduction.pdf
2. Web links and Video Lectures (e-Resources): https://cse11-iiith.vlabs.ac.in/

10.0 Magazines/Journals Used and Recommended to Students

Sl. No	Magazines/Journals	website
1	IJCOT - International Journal of Computer & Organization Trends	https://www.ieee.org/documents/ieee_focus_on_computer_hardware.pdf

11.0 Examination Note

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.

12.0 Course Delivery Plan

Module	Lecture No./Practical No	Content of Lecture	% of Portion
1	L1	Introduction to Digital Design.	20
	L2	Binary Logic, Basic Theorems And Properties Of Boolean Algebra.	
	L3	Boolean Functions.	
	L4	Digital Logic Gates, Introduction, The Map Method.	
	L5	Four-Variable Map.	
	L6	Don't-Care Conditions, NAND and NOR Implementation.	
	L7	Other Hardware Description Language.	
	L8	Verilog Model of a simple circuit.	
	P1	Given a 4-variable logic expression, simplify it using appropriate technique and simulate the same using basic gates.	
	P2	Design a 4 bit full adder and subtractor and simulate the same using basic gates.	
2	L9	Combinational Logic: Introduction.	
	L10	Combinational Circuits.	

	L11	Design Procedure, Binary Adder- Subtractor.	20
	L12	Decoders, Encoders.	
	L13	Multiplexers.	
	L14	HDL Models of Combinational Circuits – Adder, Multiplexer, Encoder.	
	L15	Sequential Logic: Introduction, Sequential Circuits.	
	L16	Storage Elements: Latches, Flip-Flops.	
	P3	Design Verilog HDL to implement simple circuits using structural, Data flow and Behavioral model.	
	P4	Design Verilog HDL to implement Binary Adder-Subtractor – Half and Full Adder, Half and Full Subtractor.	
	P5	Design Verilog HDL to implement Decimal adder.	
	P6	Design Verilog program to implement Different types of multiplexer like 2:1, 4:1 and 8:1.	
	P7	Design Verilog program to implement types of De-Multiplexer.	
	P8	Design Verilog program for implementing various types of Flip-Flops such as SR, JK and D.	
3	L17	Basic Structure of Computers: Functional Units.	20
	L18	Basic Operational Concepts, Bus structure.	
	L19	Performance – Processor Clock, Basic Performance Equation.	
	L20	Clock Rate, Performance Measurement. Machine Instructions and Programs.	
	L21	Memory Location and Addresses.	
	L22	Memory Operations.	
	L23	Instruction and Instruction sequencing.	
	L24	Addressing Modes.	
4	L25	Input/output Organization	20
	L26	Accessing I/O Devices, Interrupts.	
	L27	Interrupt Hardware, Enabling and Disabling Interrupts.	
	L28	Handling Multiple Devices, Direct Memory Access.	
	L29	Bus Arbitration, Speed, sizes.	
	L30	Cost of memory systems.	
	L31	Cache Memories.	
	L32	Mapping Functions.	
5	L33	Basic Processing Unit.	
	L34	Some Fundamental Concepts: Register Transfers.	
	L35	Performing ALU operations.	

	L36	Fetching a word from Memory.	20
	L37	Execution of a Complete Instruction. Pipelining.	
	L38	Basic concepts.	
	L39	Role of Cache memory.	
	L40	Pipeline Performance.	

13.0 QUESTION BANK

MODULE-1: Introduction to Digital Design.

1. Explain about combinational circuits and sequential circuits.(JAN 2018/ AUG2019)
2. Reduce the following using K-map technique F(A,B,C,D)=∑m(0,4,8,9,10)..(JAN 2018)
3. Find the reduced SOP form of the following function F(A,B,C,D)=∑m(0,4,8,9,10)+∑d(2,5,13).(AUG2019)
4. Explain multiplexer and demultiplexer with circuit diagrams.(JAN 2020)
5. Explain about Encoder and decoder with circuit diagrams.(JAN 2020/ AUG2019)
6. Design 5 to32 decoder using one 2 to 4 and four 3 to 8 decoder IC,s.(JAN 2020/ AUG2020)
7. Design 4 to 1 line multiplexer.(JAN 2018)
8. Design 32 to 1 multiplexer using two 74LS150.(AUG2019)
9. Design 1 to 8 multiplexer using two 1 to 4 demultiplexer.(AUG2021) .

MODULE-2: Combinational Logic.

1. What is k-map? How its simplify the Boolean expressions. .(JAN 2018)
2. What is Quine-McClusky Method? .(JAN 2019)
3. Explain Q-M method with suitable example.(JAN 2018).
4. Explain about prime implicit chart. .(JAN 2018)
5. Explain incompletely specified functions. .(JAN 2018)
6. What is prime implicants explain with suitable examples. .(JAN 2018)

MODULE-3: Basic Structure of Computers.

1. Explain with a neat diagram the connection between the processor and the computer memory. (05 Marks Jan-19)
2. Explain the Basic Instruction types with example.(05 Marks Jan-19
3. Define Addressing mode, explain the various addressing modes with example. (10 Marks Jan-19)
4. Write an assembly program that reads a line of characters and display it. (05 Marks Jan-19)
5. Point out various shifts and rotate instruction and example with a neat diagram and example. (10 Marks Jan-19)
6. With a neat diagram discuss the basic operational concept of a computer [June/July 2017]
7. Explain methods to improve the performance of computer [June/July 2017]
8. Explain Big-Endian , little Endian and assignment byte addressability [June/July 2017]
9. What are the addressing modes? Explain the different 4 types of addressing modes with examples [June/July 2017]

MODULE-4: Input/output Organization.

1. Define interrupt. Point out and explain the various ways of enabling and disabling interrupts. (07 Marks

Jan-19)

2. What are Exceptions? Point out and explain the different kinds of exceptions.(05Marks Jan-19)

3. What is interrupt nesting, explain with a neat diagram the implementation of interrupt priority, using individual interrupt request and acknowledge lines.(08 Marks Jan-19)

4. What is Bus Arbitration? Explain centralized and distributed arbitration. With a neat diagram. (10 Marks Jan-19)

5. Explain Universal serial Bus tree structure and split bus operation with a neat diagram.(10 Marks Jan-19)

6. Define bus arbitration? Explain in detail any one approach of bus arbitration. [June/July 2017]

7. What are priority interrupts? Explain any one interrupt priority scheme. [June/July 2017]

8. Write a note on register in DMA interface. [June/July 2017]\

9. With a block diagram explain how the printer interfaced to processor. [June/July 2017]

10. Explain the following with respect o U.S.B [June/July 2017

MODULE-5: Basic Processing Unit.

1.Explain synchronous DRAMS with a block diagram. (05 Marks Jan-19)

2. Define ROM; point out and explain various types of ROMS.(05 Marks Jan-19)

3. Define cache memory; explain various types of it with a neat block diagram. (10 Marks Jan-19)

4. What is Virtual memory? Explain virtual memory organization.(7 Marks Jan-19)

5. Explain the optical disk organization with a neat diagram.(10 Marks Jan-19)

6. Define Hit rate and miss penalty.(3 Marks Jan-19)

7. Define. [June/July 2017]

i) Memory Latency ii) Memory bandwidth iii) Hit-rate iv) Miss-penalty

8.With a neat diagram explain the internal organization of a 2Mx8 dynamic memory chip.[June/July 2017]

9. Explain associative mapping technique and set associative mapping technique. [June/July 2017]

10. What is virtual memory? With a diagram explain how virtual memory address is translated. [June/July 2017]

11. Write a note on [June/July 2017]

i) Magnetic tape system ii) Flash memory

14.0 University Result

Examination	S+	S /FCD	A/FC	B/sc	C	D	E	% Passing
.	.	.	.	.	.	.	.	.

Prepared by	Checked by		
Prof. N.K.Honnagoudar	Prof.M.G.Ganachari	HOD	Principal

Subject Title	**OPERATING SYSTEMS**		
Subject Code	BCS303	**CIE Marks**	50
		SEE Marks	50
Number of Lecture Hrs / Week	3:2:0:0	**Total Marks**	100
Total Number of Lecture Hrs	40T+20P	**Exam Hours**	03
Credits: 4			

FACULTY DETAILS:		
Name: Prof. Sapna Patil	**Designation:** Assistant Professor	**Experience:** 1 sem
No. of times course taught:01(including present)	**Specialization:** Digital Electronics and VLSI	

1.0 Prerequisite Subjects:

Sl. No	Branch	Semester	Subject
01	Computer Science and Engg.	I / II	Programming for problem solving

2.0 Course Objectives

- 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.

3.0 Course Outcomes

At the end of the course the students should be able to:

COs	Course Outcome	Cognitive Level	POs/PSO
C203.1	Explain the structure and functionality of operating system	L2	PO1
C203.2	Apply appropriate CPU scheduling algorithms for the given problem.	L3	PO2
C203.3	Analyze the various techniques for process synchronization and deadlock handling	L3	PO2
C203.4	Apply the various techniques for memory management	L3	PO2
C203.5	Explain file and secondary storage management strategies.	L2	PO1
C203.6	Describe the need for information protection mechanisms	L2	PO6
Total Hours of instruction		**40**	

4.0 Course Content

Module-1

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) **8 Hours**

Module-2

Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication
Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.
Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Thread scheduling; Multiple-processor scheduling,
Textbook 1: Chapter – 3 (3.1-3.4), 4 (4.1-4.4), 5 (5.1 -5.5) **8 Hours**

Module-3

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization;
Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.
Textbook 1: Chapter – 6 (6.1-6.6), 7 (7.1 -7.7) **8 Hours**

Module-4

Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.
Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.
Textbook 1: Chapter -8 (8.1-8.6), 9 (9.1-9.6) **8 Hours**

Module-5

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.
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.
Textbook 1: Chapter – 10 (10.1-10.5) ,11 (11.1-11.5),12 (12.1-12.5), 14 (14.1-14.4) **8 Hours**

5.0 Relevance to future subjects

Sl.No.	Semester	Subject	Topics
01	IV	Microcontroller	-
02	V	Data Structure	-

6.0 Relevance to Real World

Sl.No	Real World Mapping
01	Provide a platform for running application, managing hardware resources and providing a user interface.

7.0 Gap Analysis and Mitigation

Sl. No	Delivery Type	Details
01	Tutorial	Topic: Introduction, Process Management and scheduling, Memory management.
02	NPTEL	Topic: operating System Fundamentals

8.0 Books Used and Recommended to Students

Text Books

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 Dhamdhere, 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.

9.0 Relevant Websites (Reputed Universities and Others) for Notes/ Animation/ Videos Recommended

Website and Internet Contents References

1. **https://youtu.be/mXw9ruZaxzQ**
2. **https://youtu.be/vBURTt97EkA**
3. https://www.youtube.com/watch?v=783KAB-tuE4&list=PLIemF3uozcAKTgsCIj82voMK3TMR0YE_f
4. **https://www.youtube.com/watch?v=3-ITLMMeeXY&list=PL3pGy4HtqwD0n7bQfHjPnsWzkeRn6mkO**

10.0 Magazines/Journals Used and Recommended to Students

Sl. No	Magazines/Journals	Website
1	IEEE journals and magazines	https://ieeexplore.ieee.org/document/6323579
2	Journal of Operating system development and trends	https://www.magazinesubscriptions.in/Journal-of-Operating-Systems-Development-and-Trends.html
3	Evolution of Operating system	https://link.springer.com/chapter/10.1007/978-1-4757-3510-9_1

11.0 Examination Note

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.

SPECIAL INSTRUCTIONS

1. The total exam duration is 3 hours.
2. Use black ink ball point pen for writing examination.
3. Drawing should be drawn using dark pencil.
4. Read the questions carefully.
5. Answer the questions up to the point.

12.0 Course Delivery Plan

Module	Lecture No./Practical Session	Content of Lecture	% of Portion
Module-1	L1	What operating systems do, Operating System structure	20
	L2	Operating System operations Process management, Memory management, Storage management	
	L3	Protection and Security, Distributed system	
	L4	Special-purpose systems, Computing environments	
	L5	User - Operating System interface, System calls, Types of system calls	
	L6	System programs, Operating system design and implementation	
	L7	Operating System structure, Virtual machines, Operating System debugging,	
	L8	Operating System generation, System boot	
Module-2	L1	Process concept, Process scheduling	20
	L2	Operations on processes ,Inter process communication	
	L3	**Multi-threaded Programming:** Overview, Multithreading models	
	L4	Thread Libraries, Threading issues.	
	L5	**Process Scheduling**: Basic concepts, Scheduling Criteria	
	L6	Scheduling Algorithms	
	L7	Thread scheduling	
	L8	Multiple-processor scheduling	

Module-3	L1	**Process Synchronization:** Synchronization: The critical section problem; Peterson's solution	20
	L2	Classical problems of synchronization	
	L3	Synchronization hardware, Semaphores	
	L4	Classical problems of synchronization	
	L5	**Deadlocks:** System model, Deadlock characterization	
	L6	Methods for handling deadlocks	
	L7	Deadlock prevention, Deadlock avoidance	
	L8	Deadlock detection and recovery from deadlock.	
Module-4	L1	**Memory Management:** Memory management strategies	20
	L2	Background, Swapping	
	L3	Contiguous memory allocation, Paging	
	L4	Structure of page table	
	L5	Segmentation.	
	L6	**Virtual Memory Management:** Background, Demand paging	
	L7	Copy-on-write, Page replacement	
	L8	Allocation of frames, Thrashing.	
Module-5	L1	**File System, Implementation of File System:** File system: File concept, Access methods	20
	L2	Directory and Disk structure, File system mounting, File sharing	
	L3	**Implementing File system:** File system structure, File system implementation, Directory implementation;	
	L4	Allocation methods, Free space management.	
	L5	**Secondary Storage Structure, Protection:** Mass storage structures, Disk structure, Disk attachment	
	L6	Disk scheduling, Disk management	
	L7	**Protection**: Goals of protection, Principles of protection	
	L8	Domain of protection, Access matrix	

Practical Component of IPCC

SL. NO	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

13.0 QUESTION BANK

Module -1

1. Explain in detail about abstract view of the components of a computer system with a neat diagram Jan/Feb 2021 (10)
2. Explain about computer system organization with a neat diagram. Jan/Feb 2021 (10)
3. Discuss briefly about operating system operations with diagram. Jan/Feb 2021 (10)

4. What are system calls? Discuss briefly about types of system calls with illustration. Jan/Feb 2021(10)

5. Define Operating system. Explain briefly multiprogramming and time sharing system. Jan/Feb 2023 (8)

6. Explain various OS services with figure. Jan/Feb 2023 (5)

7. Explain the dual mode operation of operating system. Feb/Mar 2022 (7)

8. Explain layered approach structure of operating system with diagram. Feb/Mar 2022 (7)

9. Differentiate client server computing and peer to peer computing. Feb/Mar 2022 (6)

10. With a neat diagram, explain the concept of virtual machines. Feb/Mar 2022 (6)

11. Distinguish between following terms (10)

i) Multi programming and multitasking.

ii) Multi processor system and clustered systems. July/Aug 2022

12. Explain about system call with an example of handling a user application invoking the open() system call. July/Aug 2022 (5)

13. Explain the types of multiprocessing and types of clustering. (5) MQP 2019-20

14. What is boot strap loader? (3) MQP

15. Explain the working of system call. (4) MQP

16. How is layered structure advantageous over simple structure of operating system. MQP (4)

Module 2

1. Discuss in detail about multithreading models with suitable illustration.(10)

2. Explain about the different scheduling criteria in process scheduling concept. (10)

3. Explain in detail about multiple – processor scheduling with example (10)

4. with a neat diagram, explain the different state of process. (7)

5. Explain scheduler in a process execution. (5)

6. Explain direct and indirect communication wrt message passing system (8)

7. Discuss multithreading models. (5)

8. Discuss the benefits of multithreaded programming. (5)

9. Consider the following set of four processes with length of CPU burst given in ms (10)

Process	Burst time
P1	24
P2	3
P3	3

Compute the waiting time and avg turn around time for the above process using FCFS scheduling algorithm Jan/Feb 2023

10. Draw the Gantt chart and calculate average waiting time and turnaround time for the following snapshot of processes using i) FCFS ii) SRTF iii) RR (2ms) (7)

Process id	Burst time	Arrival time
P1	6	0
P2	3	1

P3	1	2
P4	4	3

11. Calculate the average waiting time and turnaround time for the following snapshot of process using (7)

i. Non- preemptive SJF

ii. Non – Preemptive priority

iii. Round Robin (TQ= 1 ms)

P	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

Feb/Mar 2022

12. Discuss the implementation of IPC using message passing systems in details. (10)

13. Discuss the issues that come with multithreaded programming. (10)

14. Explain CPU scheduling criteria (5)

15. Calculate the average waiting time and the average turnaround time by drawing the Gantt chart using FCFS, RR (q=2ms) and priority algorithms. Lower priority number represents higher priority. (12)

Process	Arrival Time	Burst Time	Priority
P1	0	9	3
P2	1	4	2
P3	2	9	1
P4	3	5	4

July/Aug 2022

Module 3

1. Discuss briefly about critical – section problem with example. Jan /Feb 2021

2. Illustrate with an example Peterson's solution problem.

3. Explain Dining philosopher's problem (6)

4. Show how semaphores provide solution to reader writers problems.

5. Explain critical section problem. What are the requirements that critical section problems must satisfy. Feb/Mar 2022

6. What is critical section problem? What are the requirements for the solution to critical section problem? Explain Peterson's solution. (8) July/Aug 2022

7. Discuss briefly about semaphores in synchronization.

8. Discuss in detail about deadlock characteristics with example. . Jan/Feb 2021

9. What are necessary conditions for deadlock?

10. Explain different methods to recovery from deadlocks.

11. Consider the following snapshot of system:

Process	Allocation	Maximum	Available

	A	B	C	A	B	C	A	B	C
P0	0	1	0	7	5	3	3	3	2
P1	2	0	0	3	2	2			
P2	3	0	2	9	2	2			
P3	2	1	1	2	2	2			
P4	0	0	2	4	3	3			

Find the need matrix and calculate safe sequence using bankers algorithm mention the above is safe or not safe. . Jan /Feb 2023

12. Describe the resource allocation graph i) with deadlock ii) With a cycle but no deadlock

13. Using bankers algorithm determine whether the following system is in a safer state

Process	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P0	0	0	2	0	0	4	1	0	2
P1	1	0	0	2	0	1			
P2	1	3	5	1	3	7			
P3	6	3	2	8	4	2			
P4	1	4	3	1	5	7			

If a request from process P2 arrives for (0, 0 , 2) can the request be granted immediately? Feb/Mar 2022

14. How to prevent the occurrence of deadlock, explain in detail.

15. Consider the following snapshot of a system:

Process	Allocation				Max				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	2	0	0	1	4	2	1	2	3	3	2	1
P1	3	1	2	1	5	2	5	2				
P2	2	1	0	3	2	3	1	6				
P3	1	3	1	2	1	4	2	4				
P4	1	4	3	2	3	6	6	5				

Answer the following using Banker's algorithm.

i) Is the system in safe state? If so, give the safe sequence.

ii) If process P2 request (0, 1, 1, 3) resources can it be granted immediately? Jul/Aug 2022

16. Describe the safety algorithm MQP

Module 4

1. Discuss in detail about contiguous memory allocation with illustration. Jan/Feb 2021

2. Explain in detail about paging in a memory management scheme. Jan/Feb 2021

3. What is paging hardware with TLB? Jan /Feb 2023

4. Explain structure of page table with respect to hierarchical paging. Jan /Feb 2023

5. Explain the process of segmentation. Jan /Feb 2023

6. Illustrate with example the internal and external fragmentation problem. Feb/Mar 2022

7. What are Translation load aside Buffer (TLB) ? Explain TLB in detail with a simple paging system with a neat diagram. Feb/Mar 2022

8. Explain the structure of page table. Jan/Feb 2021

9. Discuss briefly about demand- paging in memory management scheme. Jan/Feb 2021 (10)

10. Describe the steps in handling a page fault with neat diagram. Jan/Feb 2023 (06)

11. Explain copy on write process in virtual machine. Jan/Feb 2023 (06)

12. Explain FIFO and optimal page replacement algorithm. Jan/Feb 2023 (08)

13. Consider the following page reference string

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

Assuming there are 3 memory frames, how many page faults would occur in case of

i) LRU ii) Optimal algorithm note that initially all frames are empty. Feb/Mar 2022 (7)

14. Illustrate how demand paging affects system performance. Feb/Mar 2022 (07)

15. What is trashing? How it can be controlled? Feb/Mar 2022 (06)

16. Consider the page reference string: 1,0,7,1,0,2,1,2,3,0,3,2,4,0,3,6,2,1 for a memory with 3 frames. Determine the number of page faults using FIFO, optimal and LRU replacement algorithms. Which algorithm is most efficient? Jul/aug 2022 (14)

17. Explain the different frame allocation methods. Jul/aug 2022

18. Explain the basic method of implementing paging concept. MQP

19. What are the benefits of virtual memory? MQP

Module 5

1. Discuss the various directory structures with required diagrams. July/Aug 2022 (10)

2. Explain various access methods of files. Feb/Mar 2022 (07)

3. Explain the various operations performed on files. MQP (07)

4. With suitable example, explain any two methods of implementation of free space list. (06)

5. Explain with a neat diagram any two disk allocation method in detail. Jan/Feb 2023 (08)

6. List the different operations performed on a directory. Jan/Feb 2023 (06)

7. Explain tree structured directory structure. Jan/Feb 2023 (06)

8. Discuss briefly about file attributes in a file system. MQP (10)

9. Explain in detail about various file operations in a file system. MQP (10)

10. Explain in detail about various file types in a file system. MQP (10)

11. Explain in detail about overview of mass storage structure Jan/Feb 2021 (10)

12. Explain SCAN, CSCAN and LOOK scheduling techniques. Jan/Feb 2023 (08)

13. Explain access matrix model of implementing protection in OS. Jan/Feb 2023 (06)

14. Explain bad-block recovery in detail. Jan/Feb 2023 (06)

15. Explain the various disk scheduling algorithm with example. Feb/Mar 2022 (7)

16. Explain access matrix method of system protection with domain as objects and its implementation. July/Aug 2022 (10)

17. A drive has 5000 cylinders numbered 0 to 4999. The drive is currently serving a request at 143 and previously serviced a request at 125. The queue of pending requests in FIFO order is : 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130. Starting from current head position, what is the total distance travelled (in cylinders) by disk arm to satisfy the request using FCFS,

SSTF, SCAN, LOOK and C-LOOK algorithms. July/Aug 2022 (10)

14.0 University Result

Examination	FCD	FC	SC	% Passing

Prepared by	Checked by		
Prof. S. B. Patil	Prof. A. A. Daptadar	HOD	Principal

Subject Title	**DATA STRUCTURES & APPLICATIONS**		
Subject Code	BCS304	**CIE Marks**	50
		SEE Marks	50
Number of Lecture Hrs / Week	3:0:0:0	**Total Marks**	100
Total Number of Lecture Hrs	40	**Exam Hours**	03
Credits: 3			

FACULTY DETAILS:		
Name: Prof. S. I. Mane	**Designation:** Assistant Professor	**Experience: 09** yrs
No. of times course taught:02(including present)	**Specialization:** Computer Network Engineering	

1.0 Prerequisite Subjects:

Sl. No	Branch	Semester	Subject
01	Computer Science and Engg.	I / II	CProgramming for problem solving
02	Computer Science and Engg.	I / II	Matematics: Set, Relation, Matrices etc.

2.0 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 advan
ced Data structure concepts such as Hashing and Optimal Binary Search Trees

3.0 Course Outcomes

At the end of the course the students should be able to:

COs	Course Outcome	Cognitive Level	POs/PSO
C204.1	Explain different data structures and their applications.	L2	PO1, PO2, PO3, PO8, PO10, PO12, PSO1, PSO2
C204.2	Apply Arrays, Stacks and Queue data structures to solve the given problems.	L3	PO1, PO2, PO3, PO8, PO10, PO12, PSO1, PSO2
C204.3	Use the concept of linked list in problem solving.	L3	PO1, PO2, PO3, PO8, PO10, PO12, PSO1, PSO2
C204.4	Develop solutions using trees and graphs to model the real-world problem.	L3	PO1, PO2, PO3, PO8, PO10, PO12, PSO1, PSO2
C204.5	Explain the advanced Data Structures concepts such as Hashing Techniques and Optimal Binary Search Trees.	L2	PO1, PO2, PO3, PO8, PO10, PO12, PSO1, PSO2
Total Hours of instruction		**40**	

4.0 Course Content

Module-1

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 **8 Hours**
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
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 **8 Hours**
Text Book: Chapter-3: 3.3, 3.4, 3.7 Chapter-4: 4.1 to 4.4

Module-3
LINKED LISTS : Additional List Operations, Sparse Matrices, Doubly Linked List.
TREES: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees. **8 Hours**
Text Book: Chapter-4: 4.5,4.7,4.8 Chapter-5: 5.1 to 5.3, 5.5

Module-4
TREES(Cont..): Binary Search trees, Selection Trees, Forests, Representation of Disjoint sets, Counting Binary Trees,
GRAPHS: The Graph Abstract Data Types, Elementary Graph Operations **8 Hours**
Text Book: Chapter-5: 5.7 to 5.11 Chapter-6: 6.1, 6.2

Module-5
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 **8 Hours**
Text Book: Chapter 8: 8.1 to 8.3 Chapter 9: 9.1, 9.2 Chapter 10: 10.1

5.0 Relevance to future subjects

Sl.No.	Semester	Subject	Topics
01	IV	Analysisand Design of Algorithms	-
02	V/VI	S, Theory of Computation	-

6.0 evance to Real World

Sl.No	Real World Mapping
01	Implementation of solution to the problems using appropriate Data structures and algorithms.

7.0 Gap Analysis and Mitigation

Sl. No	Delivery Type	Details
01	Tutorial	Topic: Introduction, Stack and Queues, Linked List
02	NPTEL	Topic: Tress and Graphs

8.0 Books Used and Recommended to Students

Text Books
1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
Reference Books

1. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1 st Ed, McGraw Hill, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2 nd Ed, Cengage Learning,2014.
3. Reema Thareja, Data Structures using C, 3 rd 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, 2 nd Ed, PHI, 1996.

9.0 Relevant Websites (Reputed Universities and Others) for Notes/ Animation/ Videos Recommended

Website and Internet Contents References
1. http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html
2. https://nptel.ac.in/courses/106/105/106105171/
3. http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html
4. https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s
5. https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html
6. https://nptel.ac.in/courses/106/102/106102064/
7. https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html
8. https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html
9. https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html
10. https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html
11. https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html
12. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501595428077568125 59/overview

10.0 Magazines/Journals Used and Recommended to Students

Sl.No	Magazines/Journals	Website
1	ACM journals	**www.dl.acm.org/pubs.cfm**
2	IBM journal of Research & Development	**https://link.springer.com/journal/453**
3	Research papers/Data structures	**https://wiki.haskell.org/Research_papers/Data_structures**

11.0 Examination Note

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 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.

SPECIAL INSTRUCTIONS

6. The total exam duration is 3 hours.
7. Use black ink ball point pen for writing examination.
8. Drawing should be drawn using dark pencil.
9. Read the questions carefully.
10. Answer the questions up to the point.

12.0 Course Delivery Plan

Module	Lecture No./Practical Session	Content of Lecture	% of Portion
Module-1	L1	**INTRODUCTION TO DATA STRUCTURES**: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations	20
	L2	Review of pointers and dynamic Memory Allocation,	
	L3	**ARRAYS and STRUCTURES**: Arrays, Dynamic Allocated Arrays,	
	L4	Structures and Unions,	
	L5	Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings	
	L6	**STACKS**: Stacks, Stacks Using Dynamic Arrays,	
	L7	Evaluation and conversion of Expressions	
	L8	Evaluation and conversion of Expressions	
Module-2	L1	**QUEUES:** Queues,	20
	L2	Circular Queues,	
	L3	Using Dynamic Arrays,	
	L4	Multiple Stacks and queues.	
	L5	**LINKED LISTS** : Singly Linked,	
	L6	Lists and Chains, Representing Chains in C,	
	L7	Linked Stacks and Queues,	
	L8	Polynomials	
Module-3	L1	**LINKED LISTS** : Additional List Operations,.	20
	L2	Sparse Matrices,	
	L3	Doubly Linked List	
	L4	Doubly Linked List-operations	
	L5	TREES: Introduction,	
	L6	Binary Trees,	
	L7	Binary Tree Traversals,	
	L8	Threaded Binary Trees.	
Module-4	L1	**TREES(Cont..):** Binary Search trees, ,	
	L2	Selection Trees,	
	L3	Forests	
	L4	Representation of Disjoint sets,	
	L5	Counting Binary Trees,	

	L6	**GRAPHS:** The Graph Abstract Data Types,	20
	L7	Elementary Graph Operations: DFS,BFS	
	L8	Connected tree, Spanning tree	
Module-5	L1	**HASHING**: Introduction,	20
	L2	Static Hashing,	
	L3	Dynamic Hashing using Directories	
	L4	Directoryless Dynamic Hashing	
	L5	**PRIORITY QUEUES**: Single ended Priority Queues,	
	L6	double ended Priority Queues,	
	L7	Leftist Trees	
	L8	**INTRODUCTION TO EFFICIENT BINARY SEARCH TREES**: Optimal Binary Search Trees	

13.0 QUESTION BANK

Module -1

1. What is data structure? What are the various types of data structure? Explain. ***(Jan-2020/July 2022)***
2. How does a structure differ from a union? Mention any 2 uses of structure..
3. Define an array, array pointer and array of pointer. Explain declaration and initialization of 1D array and 2D array.
4. What is structure and self-referential structure? Explain with a simple example to each.
5. What is union? Explain the differences between structure and union with an example.***(July 2022)***
6. Define Data structures. Give its classification. What are the basic operations that can be performed on data structure?
7. What is structure and self-referential structure? Explain with a simple example to each.
8. Write the Knuth Morris Pratt pattern matching Algorithm/Function in C and apply the same to search the pattern : "abcdabcdy" in the text "abcxabcdabxabcdabcdabcy". ***(Jan-2020/2022)***
9. Consider the given two polynomials, $A(x) = 8x^{24} + 5x^2 + x+1$ & $B(x) = 4x^{14} + 10x^2+ 1$ Represent the polynomials using array of structures.
10. Explain with an Examples :i) malloc( ) ii) calloc( ) iii) realloc( ) iv) free( ). ***(Jan-2020/July 2022)***
11. What is the difference between Static memory allocation and dynamic memory allocation? Explain.
12. How Union is different from structure? Illustrate with example.
13. Write a C function to implement string insertion operation. Illustrate the process with suitable example.
14. Consider the given two polynomials, $A(x) = 3x^{11} + 5x^2 + 8$ and $B(x) = x^4 + 10x^2 + 1$ Represent the polynomials using array of structures.
15. Give the ADT for Sparse matrix. Express the following Sparse matrix in the triplet form and find its transpose.

$$A=\begin{pmatrix} 10 & 0 & 0 & 25 & 0 \\ 0 & 23 & 0 & 0 & 45 \\ 0 & 0 & 0 & 0 & 32 \\ 42 & 0 & 0 & 31 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 0 & 0 \end{pmatrix}$$

16. Consider the given two polynomials, $A(x) = 4x^{15} + 3x^4 + 5$ and $B(x) = x^4 + 10x^2 + 1$ Represent the polynomials using array of structures.
17. Define an array, array pointer and array of pointer. Explain declaration and initialization of 1D array and 2D array.
18. What is structure? How it is different from array? Explain different types of structure declaration with examples and give difference between Structure and Union. ***(Jan-2020)***
19. Define pointers. How to declare and initialize pointers, explain with example. ***(Jan-2020)***

20. Write a C program to: ***(Jan-2020)***
    (i). Compare strings
    (ii). Concatenate two strings
21. List and explain the functions supported in C for dynamic memory allocation. ***(Jan-2022)***
22. Explain any four string handling functions supported by C with syntax and example. ***(July-2022)***
23. How do you define a data structure? How is stack represented? Give a C program to construct a stack of integers and perform all the necessary operations on it. ***(Jan-2022)***
24. Write the algorithm to implement a stack using dynamic array whose initial capacity is 1 and array doubling is used to increase the stack's capacity whenever an element is added to a full stack. Implement the operation-Push, pop and display. ***(DEC-2016)***
25. Write an algorithm to convert a valid infix expression to a postfix expression. Also evaluate the following suffix expression for the values: A=1 B=2 C=3. AB+C-BA+C$-
26. Define stack. Implement push and pop functions for stack using arrays.
27. Write the postfix form of the following expression: (a+b)*d+e/(f+a*d)+c
28. Write the prefix form of the following expression: (a+b)*d+e/(f+a*d)+c
29. Write the postfix form of the following expression: ((a/(b-c+d))*(e-a)*c)
30. Write the prefix form of the following expression: ((a/(b-c+d))*(e-a)*c)
31. Define Stack. Give the implementation of push, pop and display functions. Include check for Empty and full conditions. ***(Jan-2020)***
32. Write the postfix form of the following expressions using stack: ***(Jan-2020)***
    (i) A $ B * C – D + E | F | (G+H)
    (ii) A – B | (C * D $ E)
33. Write an algorithm to evaluate a postfix expression and apply the same for the given postfix expression. ABC – D * + E $ F + and assume A=6, B=3, C=2, D=5 E=1 and F=7. ***(Jan-2020)***
34. Define Recursion. Write recursive functions for following: ***(Jan-2020/22)***
    (i) Factorial of a number
    (ii) Tower of Hannoi
35. Write an algorithm to convert infix expression into postfix and also trace the same for expression (a+b)*d+e/f+c. ***(Jan-2022)***
36. Write an Ackermann function and apply the same to evaluate A(1,2)***(Jan-2022)***

Module-2

1. What is the advantage of circular queue over ordinary queue? Mention any 2applications of queues. Write an algorithm CQINSERT for static implementation of circular queue.
2. Explain the working of a simple queue, dequeue and priority Queue.
3. Implement addq and deleteq functions for the circular queue.
4. What is recursion? Write a C function to find factorial and GCD.
5. What is the advantage of circular queue over ordinary queue? Write a C program to simulate the working of Circular queue of integers using array. Provide the following operations: ***(Jan-2020)***
    (i) Insert
    (ii) Delete
    (iii) Display
6. Write a note on Dequeue and priority queue. ***(Jan-2020)***
7. Explain various operations on circular queue using array.***(Jan-2022)***
8. List out any two applications of linked list and any two advantages of doubly linked list over singly linked list.
9. Write a C program to simulate an ordinary queue using a singly linked list..
10. Write a C program to simulate Stack using a singly linked list.
11. Give an algorithm to insert a node at a specified position for a given singly linked list.

12. Write a C function to add two polynomials in C and explain its memory mapping.
13. What is polynomial? What is the degree of the polynomial? Write a function to add two polynomials?
14. Write a c Program to merge two linked list.
15. Write a C program to remove duplicates in the linked list.
16. With neat Diagram, explain the circular linked list.
17. What is Linked list? Explain the different types of Linked list with neat diagram. ***(Jan-2020)***
18. Write a C function to Insert a node at front and delete a node from rear end in a circular linked list. ***(Jan-2020)***
19. Give the node structure of create a single linked list of inters and write the functions to perform the following operations: i) Create a list containing three nodes with data 10,20,30 using front insertion. Ii) Insert node with data 40 at end of list iii) Delete a node whose data is 30. iv) Display list contents. ***(Jan-2022)***

Module-3

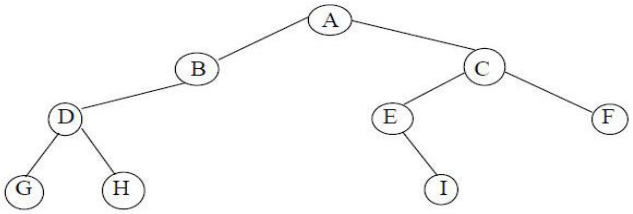
1. Write a c program to perform the following operations on doubly linked list: i) insert a node ii) delete a node.
2. For the given sparse matrix write the diagrammatic linked list representation.
3. Write a C function for the concatenation of two doubly linked lists. ***(Jan-2020)***
4. Describe the doubly linked lists with advantages and disadvantages. Write a C function to delete a node from a circular doubly linked list with header node. ***(Jan-2020)***
5. For the given sparse matrix, give diagrammatic linked representation: ***(Jan-2020)***

$$a = \begin{matrix} 0 & 1 & 2 \end{matrix} \begin{bmatrix} & & \\ 3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

6. Write a C function to add two polynomials represented as circular queue with header node. ***(Jan-2020)***
7. Write function for i) Findings the length of list ii) Concatenate two lists ii)Reverse a list. ***(Jan-2022)***
8. For the given sparse matrix, give diagrammatic linked representation ***(Jan-2022)***

$$A \begin{bmatrix} 3 & 0 & 0 & 0 \\ 5 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 8 \\ 0 & 0 & 9 & 0 \end{bmatrix}$$

9. List out differences between single linked list and double linked list .Write the functions i)Insert a node at rear end ii) Delete a node at rear end iii) search a node with given value. ***(Jan-2022)***
10. Define the following: i) Binary tree ii) Complete binary tree iii) Almost complete binary tree iv) Binary search tree v) Depth of a tree
11. Given the following graph, write the inorder, preorder and postorder traversals.



12. Construct a binary tree from the traversal order given below:
    PREORDER = A B D E F C G H L J K
    INORDER = D B F E A G C L J H K
13. Construct a expression tree for: ((6+ (3-2)*5) ^2+3) and traverse it in all 3 orders and also write their respective functions. ***(Jan-2022)***

14. What is threaded binary tree? Explain right in and left in threaded binary trees.
15. What is tree? Explain.
    i) Root node ii) degree iii) Siblings iv) Depth of the a tree and give examples. v) forest.
16. What is a binary tree? State its properties? How it is represented using array and linked list give example.
17. What is a Tree? With suitable, define: ***(Jan-2020)***
    (i) Binary tree
    (ii) Level of the Binary tree
    (iii) Complete binary tree

Module-4

1. Describe the binary search tree, with example. Write a recursive function to search for a key value in a binary Search tree.
2. Construct the b-tree from the given traversals:
   Preorder – ABDCEF In order - BDAEFC Post order – DBFECA
3. Write the C routines to traverse the tree using: ***(Jan-2020)(Jan-2022)***

(i) Pre-order traversal
(ii) Post-order traversal

4. For the given data, draw a binary search tree and show the array and linked list representation of the same: 100, 85, 45, 55, 110, 20, 70, 65. ***(Jan- 2020)***
5. What is the advantage of Threaded binary tree over binary tree? Explain the construction of threaded binary tree for 10, 20, 30, 40 and 50. ***(Jan-2020)***
6. Define expression tree. For a tree given in Fig.Q8(b) traverse the tree using in-order, pre-order and post-order traversals. ***(Jan-2020)***

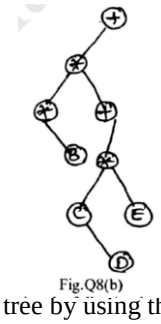


Fig.Q8(b)

7. Construct a Binary search tree by using the following in-order and pre-order traversals: ***(Jan-2020)***
   Inorder: BCAEDGHFI
   Preorder: ABCDEFGHI
8. Write the recursive search algorithm for a binary search tree. ***(Jan-2022)***
9. Define the following: i) Binary tree ii) Complete binary tree iii) Strict binary tree iv) Skewed Binary tree ***(Jan-2022)***
10. Write routines for: i) Copying of binary trees ii)Testing equality of binary trees.***(Jan-2022)***
11. Define and explain the following terminologies with suitable graph example **[July-2019]**
    i) graph ii) an edge iii) weighted edge iv) vertex v) degree of a vertex
12. Define and explain the following terminologies with suitable graph example
    i) in-degree of a vertex ii) out-degree of a vertex iii) directed graph iv) an un-directed graph
    v) complete graph vi) incomplete graph vii) sub graph viii) simple path ix) cycle
13. Explain a graph with two connected component and strongly connected component. Explain in detail the graph representations with an example
    1) Adjacency matrix
    2) Adjacency lists
    3) Adjacency multiclass
14. Explain BFS and DFS algorithms with suitable examples. ***(Jan-2022)***

15. Define graph. For the given graph, show the adjacency matrix and adjacency list representation of the graph [Ref. Fig.Q9(a)] ***(Jan-2020)(Jan-2022)***

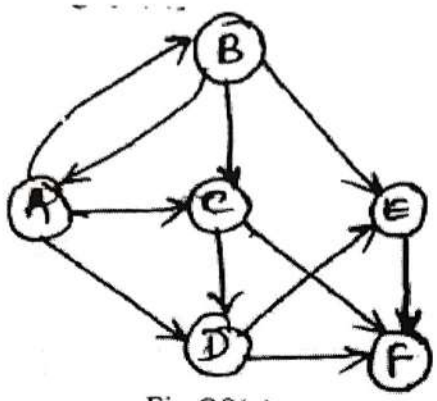


Fig.Q9(a)

16. What are the methods used for traversing a graph? Explain any one with example and write C function for the same. ***(Jan-2020)***

Module-5

1. What is hashing? Explain hash functions.***(Jan-2022)***
2. What do you mean by static hashing and dynamic hashing? Explain
3. Explain different Collision-resolution Techniques.
4. Explain Hashing and collision. What are the methods used to resolve collision. ***[July-2019]***
5. What is collision? What are the ssmethods to resolve collision? Explain linear probing with an example. ***(Jan-2020)***
6. Explain in detail about static and dynamic hashing. ***(Jan-2020)(Jan-2022)***
7. Explain optimal binary search tree.
8. Explain Single and double ended Priority Queues

14.0 University Result

Examination	FCD	FC	SC	% Passing
NA				

Prepared by	Checked by		
Prof. S. I. Mane	Dr. M. G. Huddar	HOD	Principal

Subject Title	**DATA STRUCTURES LABORATORY**		
Subject Code	BCSL305	**CIE Marks**	50
Number of Contact Hrs / Week	0:0:02	**SEE Marks**	50
Total Number of Lab Contact Hrs	28	**Exam Hours**	03
CREDITS – 01			

FACULTY DETAILS:		
Name: Prof. S. I. Mane	**Designation:**Assistant Professor	**Experience: 09** Years
No. of times course taught: 01	**Specialization:** Computer Networks Engineering	

1.0 Prerequisite Subjects:

Sl.No.	Branch	Semester	Subject
01	Computer Science & Engineering	I/II	C Programming Language

2.0 Course Objectives

This laboratory course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of

1. Dynamic memory management
2. Linear data structures and their applications such as stacks, queues and lists
3. Non-Linear data structures and their applications such as trees and graphs

3.0 Course Outcomes

Having successfully completed this course, the student will be able to:

CO	Course Outcome	Cognitive Level	POs
C205.1	Analyze various linear and non-linear data structures	L3	PO-1,2,3,5,10,12 PSO-1,2
C205.2	Demonstrate the working nature of different types of data structures and their applications	L2	PO-1,2,3,5,10,12 PSO-1,2
C205.3	Use appropriate searching and sorting algorithms for the given scenario.	L2	PO-1,2,3,5,10,12 PSO-1,2
C205.5	Apply the appropriate data structure for solving real world problems	L3	PO-1,2,3,5,10,12 PSO-1,2
Total Hours of instruction		**28**	

4.0 Course Content

Description (If any):

- Implement all the programs in "C " Programming Language and Linux OS.

Laboratory Experiments:

1. Develop a Program in C for the following:

   a) 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).

b) 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. Develop a Program in C for the following operations on Strings.

   a. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP)

   b. 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 Support the program with functions for each of the above operations. Don't use Built-in functions.

3. Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)

   a. Push an Element on to Stack

   b. Pop an Element from Stack

   c. Demonstrate how Stack can be used to check Palindrome

   d. Demonstrate Overflow and Underflow situations on Stack

   e. Display the status of Stack

   f. Exit

   Support the program with appropriate functions for each of the above operations

4. 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.

5. Develop a Program in C for the following Stack Applications a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ b. Solving Tower of Hanoi problem with n disks.

6. Develop a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)

   a. Insert an Element on to Circular QUEUE

   b. Delete an Element from Circular QUEUE

   c. Demonstrate Overflow and Underflow situations on Circular QUEUE

   d. Display the status of Circular QUEUE

   e. Exit

   Support the program with appropriate functions for each of the above operations.

7. Develop a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Programme, Sem, PhNo

   a. Create a SLL of N Students Data by using front insertion.

   b. Display the status of SLL and count the number of nodes in it

   c. Perform Insertion / Deletion at End of SLL

   d. Perform Insertion / Deletion at Front of SLL(Demonstration of stack)

   e. Exit.

8. Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo

   a. Create a DLL of N Employees Data by using end insertion.

   b. Display the status of DLL and count the number of nodes in it

   c. Perform Insertion and Deletion at End of DLL

   d. Perform Insertion and Deletion at Front of DLL

   e. Demonstrate how this DLL can be used as Double Ended Queue.

f. Exit

9. Develop a Program in C for the following operationson Singly Circular Linked List (SCLL) with header nodes

a. Represent and Evaluate a Polynomial P(x,y,z) = 6x 2 y 2 z-4yz 5 +3x 3 yz+2xy 5 z-2xyz 3

b. Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z) Support the program with appropriate functions for each of the above operations.

10. Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers .

a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2

b. Traverse the BST in Inorder, Preorder and Post Order

c. Search the BST for a given element (KEY) and report the appropriate message

d. Exit.

11. Develop a Program in C for the following operations on Graph(G) of Cities

a. Create a Graph of N cities using Adjacency Matrix.

b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method

12. 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: K →L as H(K)=K mod 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.

5.0 Relevance to future subjects

Sl. No	Semester	Subject	Topics
1	IV	Design and Analysis of Algorithms	Design and develop the programs

6.0 Relevance to Real World

Sl.No	Real World Mapping
01	Apply the knowledge to design and develop the software applications

7.0 Gap Analysis and Mitigation

Sl. No.	Delivery Type	Details
01	YouTube Videos	Design and Develop programs for linear and non-linear data structures

8.0 Books Used and Recommended to Students

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 1 st Ed, McGraw Hill, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2 nd Ed, Cengage Learning,2014.
3. Reema Thareja, Data Structures using C, 3 rd 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, 2 nd Ed, PHI, 1996

9.0 Relevant Websites (Reputed Universities and Others) for Notes/Animation/Videos Recommended

Website and Internet Contents References
1. http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html
2. https://nptel.ac.in/courses/106/105/106105171/
3. http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html
4. https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s
5. https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html

10.0 Magazines/Journals Used and Recommended to Students

Sl.No	Magazines/Journals	Website
1	Research papers/Data structures	https://wiki.haskell.org/Research_papers/Data_structures

11.0 Examination Note

- 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)

c) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks

d) For laboratories having PART A and PART B

i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks

ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks

12.0 Course Delivery Plan

Expt. No.	Lab. No.	Content of Experiment	% of Portion
1	1	Develop a Program in C for the following: a) 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). b) 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.	8.33
2	**2**	Develop a Program in C for the following operations on Strings. a. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) b. 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 Support the program with functions for each of the above operations. Don't use Built-in functions.	8.33

3	**3**	Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) a. Push an Element on to Stack b. Pop an Element from Stack c. Demonstrate how Stack can be used to check Palindrome d. Demonstrate Overflow and Underflow situations on Stack e. Display the status of Stack f. Exit Support the program with appropriate functions for each of the above operations	8.33
4	**4**	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.	8.33
5	**5**	Develop a Program in C for the following Stack Applications a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ b. Solving Tower of Hanoi problem with n disks.	8.33
6	**6**	Develop a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX) a. Insert an Element on to Circular QUEUE b. Delete an Element from Circular QUEUE c. Demonstrate Overflow and Underflow situations on Circular QUEUE d. Display the status of Circular QUEUE e. Exit Support the program with appropriate functions for each of the above operations.	8.33
7	**7**	Develop a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Programme, Sem, PhNo a. Create a SLL of N Students Data by using front insertion. b. Display the status of SLL and count the number of nodes in it c. Perform Insertion / Deletion at End of SLL d. Perform Insertion / Deletion at Front of SLL(Demonstration of stack) e. Exit.	8.33
8	**8**	Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo a. Create a DLL of N Employees Data by using end insertion. b. Display the status of DLL and count the number of nodes in it c. Perform Insertion and Deletion at End of DLL d. Perform Insertion and Deletion at Front of DLL e. Demonstrate how this DLL can be used as Double Ended Queue. f. Exit	8.33
9	**9**	Develop a Program in C for the following operationson Singly Circular Linked List (SCLL) with header nodes a. Represent and Evaluate a Polynomial P(x,y,z) = 6x 2 y 2 z-4yz 5 +3x 3 yz+2xy 5 z-2xyz 3 b. Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z) Support the program with appropriate functions for each of the above	8.33

		operations.	
10	**10**	Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers . a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 b. Traverse the BST in Inorder, Preorder and Post Order c. Search the BST for a given element (KEY) and report the appropriate message d. Exit.	8.33
11	**11**	Develop a Program in C for the following operations on Graph(G) of Cities a. Create a Graph of N cities using Adjacency Matrix. b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method	8.33
12	**12**	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: K →L as H(K)=K mod 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.	8.33

13.0 QUESTION BANK

1. What is data structure?
2. What is a sequential file and how to create it?
3. What are the functions used for handling files?
4. What are the string handlings functions?
5. Differentiate structure and union.
6. What is stack and how to implement it?
7. Differentiate infix, postfix and prefix expressions.
8. How to evaluate postfix expression using stack.
9. What is queue and how to implement it?
10. What is circular queue?
11. What are pointer and pointer to pointer?
12. What is dynamic memory allocation?
13. What is singly linked list and how to construct it?
14. What is doubly linked list and how to implement it?
15. What is binary search tree and how to traverse inorder, preorder and postorder?
16. What is recursion?
17. Explain the working of binary search technique.
18. What is tree?
19. What is min heap?
20. Illustrate max heap with an example.
21. What is graph?
22. What is an array?
23. What is priority queue?
24. What is advantage of priority queue over simple queue?
25. What are the applications of stack?
26. What are the applications of queue?
27. What is thread?
28. What is sparse matrix?
29. Explain the advantages and disadvantages of sparse matrix.
30. What is meant by hashing in data structure?

14.0 University Result

Examination	FCD	FC	SC	FAIL	% Passing
NA					

Prepared by	Checked by		
Prof. S. I. Mane	Dr. M. G. Huddar	HOD	Principal

Subject Title	**OBJECT ORIENTED PROGRAMMING WITH JAVA**		
Subject Code	BCS306A	**CIA Marks**	50
No of Lecture Hrs + Practical Hrs/ Week	2:0:2	**SEE Marks**	50
Total No of Lecture + Practical Hrs	28T + 20P	**Total Marks**	100
Examination Type (SEE)	Theory	**Exam Hours**	03
CREDITS – 03			

FACULTY DETAILS:		
Name: Prof.Prasanna Patil	**Designation:** Asst.Professor	**Experience:10**Years
No. of times course taught:03 Time	**Specialization:**Computer Science & Engineering	

1.0 Prerequisite Subjects:

Sl. No	Branch	Semester	Subject
01	Computer Science and Engineering	I/II	Programming in C
02	Computer Science and Engineering	III	Programming in C++

2.0 Course Objectives

1. To learn primitive constructs JAVA programming language.
2. To understand Object Oriented Programming Features of JAVA.
3. To gain knowledge on: packages, multithreaded programing and exceptions.

3.0 Course Outcomes

The student, after successful completion of the course, will be able to

COs	Course Outcome	RBT Level	POs
C206.1	Demonstrate proficiency in writing simple programs involving branching and looping structures.	**L3**	**1,2,3,5,8,9,10,11,12**
C206.2	Design a class involving data members and methods for the given scenario.	**L3**	**1,2,3,5,8,9,10,11,12**
C206.3	Apply the concepts of inheritance and interfaces in solving real world problems.	**L3**	**1,2,3,5,8,9,10,11,12**
C206.4	Use the concept of packages and exception handling in solving complex problem.	**L3**	**1,2,3,5,8,9,10,11,12**
C206.5	Apply concepts of multithreading, autoboxing and enumerations in program development.	**L3**	**1,2,3,5,8,9,10,11,12**
	Total Hours of instruction	**28**	

4.0 Course Content

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

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.

Interfaces: Interfaces, Default Interface Methods, Use static Methods in an Interface, Private Interface Methods.

Chapter 8, 9

Module-4

Packages: Packages, Packages and Member Access, Importing Packages.

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.

Chapter 9, 10

Module-5

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.

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).

Chapter 11, 12

5.0 Relevance to future subjects

SL. No	Semester	Subject	Topics / Relevance
01	VI	Project work	Academic Mini Project
02	VIII	Project work	Academic Project

6.0 Relevance to Real World

SL.No	Real World Mapping
01	Development of Android Applications.
02	Development of Database Applications using Java.

7.0 Books Used and Recommended to Students

Text Books
1. Java: The Complete Reference, twelth 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)
Additional Study material & e-Books
1. Java Notes for Professionals. Download: https://goalkicker.com/JavaBook/

2. Object-Oriented vs. Functional Programming. Download: http://www.oreilly.com/programming/free/object-oriented-vs-functional-programming.csp
3. Java: The Legend. Download: http://www.oreilly.com/programming/free/java-the-legend.csp

8.0 Relevant Websites (Reputed Universities and Others) for Notes/Animation/Videos Recommended

Website and Internet Contents References
1. https://www.coursera.org/
2. https://swayam.gov.in/
3. https://nptel.ac.in/
4. https://www.udemy.com/
5. https://www.mooc.org/
6. https://www.geeksforgeeks.org/java/
7. https://ocw.mit.edu/courses/6-092-introduction-to-programming-in-java-january-iap-2010/
8. https://www.w3schools.com/java/
9. https://www.javatpoint.com/java-tutorial

9.0 Magazines/Journals Used and Recommended to Students

Sl.No	Magazines/Journals	website
1	CSI communications	www.csi-india.org
2	Java Magazine - Oracle	www.oracle.com/technetwork/java/javamagazine/
3	Java - IEEE Conferences, Publications, and Resources	https://www.computer.org/software-magazine/
4	Java Developer's Journal - Steven Gould	https://jserd.springeropen.com/

10.0 Examination Note

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.

11.0 Course Delivery Plan

Module	Lecture No.	Content of Lecturer	% of Portion
MODULE 1	1	Introduction to Programming Languages	20%
	2	Object-Oriented Programming, Using Blocks of Code, Lexical Issues	
	3	The Primitive Types, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions	
	4	Arrays, Introducing Type Inference with Local Variables.	
	5	Arithmetic Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses.	
	6	Java's Selection Statements, Iteration Statements, Jump Statements	
	P1	Develop a JAVA program to add TWO matrices of suitable order N (The value of N should be read from command line arguments).	
	P2	Develop a stack class to hold a maximum of 10 integers with suitable methods. Develop a JAVA main method to illustrate Stack operations	
MODULE 2	7	Class Fundamentals,	20%
	8	Declaring Objects, Assigning Object Reference Variables,	
	9	Introducing Methods, Constructors, The this Keyword, Garbage Collection.	
	10	Overloading Methods, Objects as Parameters	
	11	Argument Passing, Returning Objects, Recursion	
	12	Access Control, Understanding static, Introducing final, Introducing Nested and Inner Classes.	
	P3	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. 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 getXY() 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	

	P4	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.**	
	P8	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.	
MODULE 3	13	Inheritance: Inheritance Basics, Using super	20%
	14	Creating a Multilevel Hierarchy	
	15	When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch	
	16	Using Abstract Classes, Using final with Inheritance, Local Variable Type Inference and Inheritance, The Object Class.	
	17	Interfaces: Interfaces, Default Interface Methods	
	18	Use static Methods in an Interface, Private Interface Methods.	
	P5	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(). Demonstratepolymorphism concepts by developing suitable methods, defining member data and main program.	
	P6	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 implementthe respective methods to calculate the area and perimeter of each shape.	
	P7	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.	
MODULE 4	19	Packages: Packages, Packages and Member Access, Importing Packages.	20%
	20	Exceptions: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch	
	21	Multiple catch Clauses, Nested try Statements, throw, throws, finally	
	22	Java's Built-in Exceptions	
	23	Creating Your Own Exception Subclasses, Chained Exceptions.	
	P9	Develop a JAVA program to raise a custom exception (user defined exception) for DivisionByZero using try, catch, throw and finally.	
	P10	Develop a JAVA program to create a package named mypack and import & implement it in a suitable class.	
MODULE 5	24	Multithreaded Programming: The Java Thread Model, The Main Thread	20%
	25	Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities	
	26	Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads, Obtaining a Thread's State.	
	27	Enumerations, Type Wrappers and Autoboxing: Enumerations	
	28	Type Wrappers, Autoboxing	
	P11	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).	
	P12	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.	

12.0 Assignments, Pop Quiz, Mini Project, Seminars

Sl. No.	Title	Outcome expected	Allied study	Week No.	Individual / Group activity	Reference: book/website /Paper
1	Assignment 1: Some important University Questions on Module one.	Students study the Topics and write the Answers. Get practice to solve questions.	Module Four of the syllabus	4	Individual Activity. Witten solutions expected.	Text book
2	Assignment 2: Some important University Questions on Module two	Students study the Topics and write the Answers. Get practice to solve questions.	Module Five of the syllabus	9	Individual Activity. Witten solutions expected.	Text book

13.0 Question Bank

Module 1:

1. Explain the features of Java.
2. Elucidate how Java is a platform independent language, with neat sketches.
3. List and explain Java buzzwords.
4. Explain the process of creating and running Java programs.
5. Explain the structure of a Java program and its keywords with an example.
6. Write & demonstrate a Java program to initialize & display different types of integers & floating type variables.
7. Explain different access specifiers in Java & their scope.
8. Define type casting. Explain with an example.
9. Explain type conversion, with an example.
10. What is type casting? Illustrate with an example. What is meant by automatic type promotion?
11. How are arrays defined in Java? Explain with an example.
12. Discuss operators in Java.
13. What is a jump statement? Explain with examples.
14. Explain: i) >>> ii)short circuit logical operators iii) for each
15. With an example explain the working of >> and >>> (unsigned right shift)
16. Write a Java program to print the factorial of the number 'n' using the "for" loop.
17. Write a program to calculate the average among the elements {8, 6, 2, 7} using "for each" in Java. Howis "for each" different from "for" loop?
18. Write a Java program to sum only the first five elements of the array {1,2,3,4,5,6,7,8,9,10} using "foreach" version of the for loop.
19. Write a java program to sum only first five elements of the array using for each looping.
20. Explain the operation of the following operators with examples. i) % ii) >>> iii) &&
21. How to declare two dimensional arrays in java? Explain with a simple example.
22. Write a Java program to illustrate the use of multidimensional arrays.

Module 2:

1. Define a class in Java and explain its role in object-oriented programming. Discuss the components that constitute a class.
2. Explain the significance of access modifiers (public, private, protected, default) in Java classes. Describe their impact on the accessibility of class members.
3. Discuss the concept of constructors in Java classes. Explain their purpose, types, and rules for constructor invocation.
4. Differentiate between instance variables and class variables in Java. Provide examples illustrating both types of variables and their usage.
5. Describe the concept of inheritance in Java classes. Explain how inheritance promotes code reusability

and the 'is-a' relationship between classes.

6. Define methods in Java and explain their role in classes. Discuss the components of a method signature.
7. Explain the differences between instance methods and static methods in Java. Provide examples illustrating when each type of method is appropriate.
8. Discuss method overloading and method overriding in Java. Provide examples demonstrating these concepts and explain their significance in inheritance.
9. Explain the return types of methods in Java, including void and non-void return types. Discuss the implications of these return types in method implementation.
10. Describe the concept of method parameters in Java. Explain the differences between pass-by-value and pass-by-reference, and how Java handles method arguments.
11. Discuss the importance of encapsulation, inheritance, and polymorphism in Java classes and methods. Explain how they contribute to robust object-oriented programming.
12. Explain the concept of a constructor in Java. Describe its role in initializing objects and discuss constructor overloading.
13. Discuss the concept of method visibility and access modifiers in Java. Explain the differences between public, private, protected, and default access modifiers for methods.
14. Explain the concept of method overloading and provide examples demonstrating its usage in Java. Discuss the rules that govern method overloading.
15. Describe the purpose and usage of the 'this' keyword in Java classes and methods. Provide examples demonstrating its use in different contexts.

Module 3:

1. Define inheritance. List the different types of inheritance. (Jan-2018)
2. Define inheritance in object-oriented programming. Explain its significance and how it promotes code reusability.
3. Differentiate between single inheritance and multiple inheritance. Provide examples illustrating both types.
4. What is the purpose of the 'super' keyword in inheritance? Provide an example demonstrating its usage in a programming language of your choice.
5. Discuss the following terms with an example: i) super ii) final (Jan-2019)
6. Explain the concepts of base class and derived class in inheritance. Provide an example illustrating both classes and their relationship.
7. Discuss the advantages and disadvantages of inheritance in object-oriented programming. Include examples to support your points.
8. Describe the terms 'method overriding' and 'method overloading' in the context of inheritance. Provide examples to illustrate each concept.
9. How does inheritance promote the concept of polymorphism in object-oriented programming? Explain with suitable examples.
10. Explain the 'is-a' relationship and the 'has-a' relationship in the context of inheritance. Provide examples to clarify the differences between these relationships.
11. Discuss the concept of access specifiers (public, private, protected) in inheritance. How do they affect the accessibility of members in classes?
12. Elaborate on the concept of a constructor in inheritance. How are constructors inherited in derived classes? Provide examples to explain.
13. When constructors are called in the class hierarchy?
14. Define what an **interface** is in object-oriented programming. Explain its purpose and how it differs from a class.
15. Discuss the importance of interfaces in achieving abstraction and code reusability in programming.
16. Differentiate between interfaces and abstract classes. When would you prefer using an interface over an abstract class, and vice versa?
17. Explain the concept of multiple interface implementation in a programming language of your choice. Provide an example illustrating this concept.
18. Write a Java program to define an interface called Area which contains method called Compute( ) and calculate the areas of rectangle (ℓ * b) and triangle (1/2 * b * h) using classes Rectangle and Triangle.

19. How do interfaces support multiple inheritance in languages that do not allow multiple inheritance of classes? Explain with examples.
20. Discuss the role of access modifiers (public, private, protected) within interfaces. Can interfaces contain member variables, and if so, what are the access modifiers for these variables?
21. Elaborate on the use of default and static methods in interfaces. Provide scenarios where each type of method would be beneficial.
22. Explain the significance of interface implementation for achieving loose coupling and flexibility in software design.
23. Describe how interfaces facilitate polymorphism in object-oriented programming. Provide examples demonstrating polymorphic behavior using interfaces.
24. Discuss real-world examples or scenarios where interfaces are particularly useful in software development, emphasizing their advantages and applications.

Module 4:

1. Explain the package and its types and import commands in Java with examples.
2. Describe the various levels of access protections available for packages and their implications.
3. Which is the alternative method to implement multiple inheritance in Java? Explain with an example.
4. Explain the role of interfaces while implementing multiple inheritance in Java.
5. Give the basic form of an exception handling block.
6. Define the role of Exception handling in software development.
7. What is an exception? Give an example for nested try statements.
8. Define exceptions. Explain the exception handling mechanism with an example.
9. Explain Java's built-in exceptions.
10. What is the importance of the clause finally?
11. Create a try block that is likely to generate three types of exception and incorporate necessary catch block to catch and handle them.
12. Write a Java program for illustrating the exception handling when a number is divided by zero and an array has a negative index value.

Module 5:

1. Define multithreading and explain its significance in computer programming. Highlight the differences between multithreading and multiprocessing.
2. Describe the life cycle of a thread in a multithreaded environment. Explain the states a thread can be in and the transitions between these states.
3. What is the difference between a process and a thread? Explain how multithreading differs from multitasking and how threads within a process share resources.
4. Discuss the advantages and challenges of multithreaded programming. Provide examples to illustrate scenarios where multithreading is beneficial and situations where it might introduce complexities.
5. Explain the concept of thread synchronization. Discuss common synchronization mechanisms used in multithreaded programming, such as locks, mutexes, semaphores, and monitors.
6. Describe the terms 'thread safety' and 'race condition.' Explain why race conditions occur in multithreaded programs and how they can be mitigated.
7. Discuss the role of critical sections in multithreaded programming. Explain how critical sections help prevent race conditions and ensure data integrity.
8. Explain the difference between user-level threads and kernel-level threads. Discuss their advantages, disadvantages, and scenarios where each type might be more suitable.
9. Describe deadlock and starvation in the context of multithreading. Provide examples and explain strategies to prevent or mitigate these issues.
10. Discuss the use of thread pools in multithreaded programming. Explain their purpose, advantages, and how they optimize resource management.
11. Define **enumerations** in programming. Explain their purpose and advantages compared to using constants or variables.
12. Discuss the key features and characteristics of enumerations in a programming language of your choice. Provide examples demonstrating the syntax and usage of enums.

13. Explain the concept of enum constants and how they differ from regular variables. Describe scenarios where using enums would be preferable over other data structures.
14. Discuss the role of methods and properties within enumerations. Provide examples showcasing methods and properties associated with enums.
15. How do enumerations enhance code readability and maintainability? Provide examples illustrating how enums improve code quality.
16. Define **type wrappers** (also known as wrapper classes) and their purpose in programming languages. Discuss why they are used and their relationship with primitive data types.
17. Explain the necessity of wrapper classes for primitive types. Provide examples demonstrating how to create instances of wrapper classes and convert between primitive types and their corresponding wrappers.
18. Discuss the methods and functionalities provided by wrapper classes that are not available with primitive data types. Illustrate these functionalities with examples.
19. Describe scenarios where using wrapper classes is advantageous, such as when working with collections or APIs that require objects rather than primitives.
20. Explain the concept of autoboxing and unboxing in relation to wrapper classes. Provide examples demonstrating autoboxing and unboxing operations in code.
21. Define **autoboxing** and explain how it simplifies code in programming languages that support this feature. Discuss its benefits and potential drawbacks.
22. Explain the mechanism of autoboxing and unboxing in detail. How does the language handle automatic conversion between primitives and their corresponding wrapper classes?
23. Discuss scenarios where autoboxing can lead to unexpected behavior or performance issues. Provide suggestions on when to avoid excessive use of autoboxing.
24. Describe the impact of autoboxing on code readability and performance. Explain how it affects memory usage and execution efficiency.
25. Provide examples illustrating the differences in code before and after the introduction of autoboxing, showcasing its impact on code clarity and simplicity.

14.0 University Result

Examination	# of Students Appeared	# of Students Passed	FCD	FC	SC	Fail	% Passing

Prepared by	Checked by		
Prof.Prasanna Patil	Dr. Mahesh Huddar	HOD	Principal

SUBJECT TITLE	**Data Visualization with Python Lab**		
Subject Code	**BCS358D**	**IA Marks**	50
Number of Contact Hours/Week	**0: 0: 2: 0**	**Exam Hours**	02
Total Number of Lab Contact Hours	**02**	**Exam Marks**	50
CREDITS - 01			

FACULTY DETAILS:		
Name: Prof N. K. Honnagoudar	**Designation:** Asst. Professor	**Experience: 21** Years
No. of times course taught: 01 Div:B	**Specialization:** Electronics	

1.0 Prerequisite Subjects:

Sl. No.	Branch	Semester	Subject
01	Computer Science & Engineering	I/II	Computer programming using C Language.

2.0 Course Objectives

This laboratory courses enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of

1. Demonstrate the use of IDLE or PyCharm IDE to create Python Applications .
2. Use Python programming constructs to develop programs for solving real-world problems.
3. Use Matplotlib for drawing different Plots.
4. Demonstrate working with Seaborn, Bokeh.
5. Working with Plotly for 3D, Time Series and Maps.

3.0 Course Outcomes

The student, after successful completion of the course, will be able to

CO	Course Outcome	RBT Level	POs
C208.1	Demonstrate the use of IDLE or PyCharm IDE to create Python Applications.	L3	1,2,3,4,6,8,12
C208.2	Use Python programming constructs to develop programs for solving real-world problems.	L3	1,2,3,4,6,8,12
C208.3	Use Matplotlib for drawing different Plots.	L3	1,2,3,4,6,8,12
C208.4	Demonstrate working with Seaborn, Bokeh for visualization	L3	1,2,3,4,6,8,12
C208.5	Use Plotly for drawing Time Series and Maps.	L3	1,2,3,4,6,8,12
Total Hours of instruction		100	

4.0 Course Content

Laboratory Experiments

Lab No.	Content of Experiment PART- A (Data visualization with python)
1	a) Write a python program to find the best of two test average marks out of three test's marks accepted from the user. b) Develop a Python program to check whether a given number is palindrome or not andalso count the number of occurrences of each digit in the input number.
2	) Defined as a function F as Fn = Fn-1 + Fn-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. b) Develop a python program to convert binary to decimal, octal to hexadecimal using functions.
3	a) Write a Python program that accepts a sentence and find the number of words, digits, uppercase letters and lowercase letters. b) Write a Python program to find the string similarity between two given strings Sample Output: Sample Output: Original string: Original string: Python Exercises Python Exercises Similarity between two said strings: Similarity between two said Strings: strings:1.0 0.967741935483871 Strings
4	a) Write a Python program to Demonstrate how to Draw a Bar Plot using Matplotlib. b) Write a Python program to Demonstrate how to Draw a Scatter Plot using Matplotlib.
5	a) Write a Python program to Demonstrate how to Draw a Histogram Plot using Matplotlib. b) Write a Python program to Demonstrate how to Draw a Pie Chart using Matplotlib.
6	a) Write a Python program to illustrate Linear Plotting using Matplotlib. b) Write a Python program to illustrate liner plotting with line formatting using matlib.
7	Write a Python program which explains uses of customizing seaborn plots with Aesthetic functions
8	Write a Python program to explain working with bokeh line graph using Annotations and Legends. a) Write a Python program for plotting different types of plots using Bokeh.
9	Write a Python program to draw 3D Plots using Plotly Libraries.
10	a) Write a Python program to draw Time Series using Plotly Libraries. b) Write a Python program for creating Maps using Plotly Libraries.

5.0 Relevance to future subjects

Sl. No	Semester	Subject	Topics / Relevance
01	I/II	Basic C languages	Programming writing skill.

6.0 Relevance to Real World

Sl. No	Real World Mapping
01	Apply the knowledge to Design and develop the programmes in various languages.

7.0 Books Used and Recommended to Students

Text Books 1. Al Sweigart, "Automate the Boring Stuff with Python",1stEdition, 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" 1 st Edition, O'REILLY.**
Additional Study material & e-Books
1. CSE Department DVP lab manual.

8.0 Relevant Websites (Reputed Universities and Others) for Notes/Animation/Videos Recommended

Website and Internet Contents References
3. www.iitg.ac.in/apvajpeyi/ph218/PH-218%20-%20Introduction.pdf
4. https\\ www.youtube.com/watch?v=_-uQrJOTkZlc

9.0 Magazines/Journals Used and Recommended to Students

Sl. No	Magazines/Journals	Website
1	ACM journals	www.dl.acm.org/pubs.cfm
2	IBM journal of Research and Development	https://link.springer.com/journal/453
3	Research papers/MIML	https://wiki.haskell.org/Research_papers/ *Machine learning using python*

10.0 Examination Note

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

11.0 Course Delivery Plan

Expt. No.	Lab No.	Content of Experiment PART- A (Data visualization with python)	% of Portion
1	1	a) Write a python program to find the best of two test average marks out of three test's marks accepted from the user. 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. Datatypes:https://www.youtube.com/watch?v=gCCVsvgR2KOperators: https://www.youtube.com/watch?v=v5MR5JnKcZIFlowControl: https://www.youtube.com/watch?v=PqFKRqpHrjwForloop: https://www.youtube.com/watch?v=0ZvaDa8eT5sWhileloop: https://www.youtube.com/watch?v=HZARImviDxgExceptions: https://www.youtube.com/watch?v=6SPDvPK38tw	10
2	2	a) Defined as a function F as Fn = Fn-1 + Fn-2. Write a Python program which accepts	20

		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. b) Develop a python program to convert binary to decimal, octal to hexadecimal using functions. 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	
3	3	a) Write a Python program that accepts a sentence and find the number of words, digits, uppercase letters and lowercase letters. b) Write a Python program to find the string similarity between two given strings Sample Output: Original string: Python Exercises Similarity between two said strings: Strings: https://www.youtube.com/watch?v=lSItwlnF0eU Sample Output: Original string: Python Exercises Similarity between two said strings:1.0 0.967741935483871 Strings functions: https://www.youtube.com/watch?v=9a3CxJyTq00	30
4	4	a) Write a Python program to Demonstrate how to Draw a Bar Plot using Matplotlib. b) Write a Python program to Demonstrate how to Draw a Scatter Plot using Matplotlib. https://www.youtube.com/watch?v=RRHQ6Fs1b8w&list=PLjVLYmrlmjGcC0B_FP3bkJ□JIPkV5GuZR&index=3 https://www.youtube.com/watch?v=7ABCuhWO9II&list=PLjVLYmrlmjGcC0B_FP3bkJ□JIPkV5GuZR&index=4	40
5	5	a) Write a Python program to Demonstrate how to Draw a Histogram Plot using Matplotlib. b) Write a Python program to Demonstrate how to Draw a Pie Chart using Matplotlib. https://www.youtube.com/watch?v=Qk7caotaQUQ&list=PLjVLYmrlmjGcC0B_FP3bkJ□JIPkV5GuZR&index=6 https://www.youtube.com/watch?v=PSji21jUNO0&list=PLjVLYmrlmjGcC0B_FP3bkJ□JIPkV5GuZR&index=	50
6	6	a) Write a Python program to illustrate Linear Plotting using Matplotlib. b) Write a Python program to illustrate liner plotting with line formatting using Matplotlib. https://www.youtube.com/watch?v=UO98lJQ3QGI&list=PL-osiE80TeTvipOqomVEeZ1HRrcEvtZB	60
7	7	Write a Python program which explains uses of customizing seaborn plots with Aesthetic functions. https://www.youtube.com/watch?v=6GUZXDef2U0	70
8	8	Write a Python program to explain working with bokeh line graph using Annotations and Legends. a) Write a Python program for plotting different types of plots using Bokeh. https://www.youtube.com/watch?v=HDvxYoRadcA	80
9	9	Write a Python program to draw 3D Plots using Plotly Libraries. https://www.youtube.com/watch?v=cCck7hCanpw&list=PLE50-dh6JzC4onX□qkv9H3HtPb	90
10	10	a) Write a Python program to draw Time Series using Plotly Libraries. b) Write a Python program for creating Maps using Plotly Libraries. https://www.youtube.com/watch?v=xnJ2TNrGYik&list=PLE50-dh6JzC4onX□qkv9H3HtPbBVA8M94&index=5 https://www.youtube.com/watch?v=D35m2CdMhVs&list=PLE50-dh6JzC4onX□qkv9H3HtPbBVA8M94&index=6	100

12.0 ORAL QUESTION BANK

1. What is a program.
2. Differentiate between python and C programming.
3. What are the applications of python?
4. Define debugging.
5. What is script mode.
6. What is function.
7. Write the structure of the function in python.
8. What is encapsulation.
9. What is chained conditionals.
10. What is traversing list.
11. What is dictionaries.
12. With example explain looping and dictionaries.
13. What is global variables.
14. What is classes and objects.
15. What is attributes.
16. Explain about prototyping verses planning.
17. Explain about classes and methods.
18. Explain about operator overloading.
19. What is polymorphism.
20. Explain about inheritance.
21. Explain parsing html.
22. What is web service.
23. What is parsing JSON.
24. Explain about object life cycle.
25. What is SQL.

13.0 University Result

Examination	S+	S/FCD	A/FC	B/SC	C	D	E	% Passing

Prepared by	Checked by		
			
Prof. N. K.Honnagoudar	Dr. K .B. Manwade	HOD	Principal