

NATIONAL INSTITUTE OF TECHNOLOGY PATNA

End Semester Exam, July- December 2023

Social and Professional Ethics (HS34151)

B.Tech.- M. Tech. DD- CSE- DS- Sem. 3

Time: 3 hours

M.M. 60

Note: Answer the following questions in maximum 500- 600 words each. Each question carries equal marks.

1. What do you understand by the word character and personality? What measures you have taken during the course learning (Social and Professional Ethics) to improve your character and personality?
2. Discuss the notion of spirituality with reference to Swami Vivekanand life journey.
3. Write down a short essay on life sketch of Pandit Madan Mohan Malviya.
4. What do you mean by corruption? Is it a social problem or individual problem? Suggest the remedial ways to check the problem of corruption.

नोट: निम्नलिखित प्रत्येक प्रश्न का उत्तर अधिकतम 500- 600 शब्दों में दें। प्रत्येक प्रश्न समान अंक का है।

1. चरित्र एवं व्यक्तित्व शब्द से आप क्या समझते हैं? आपने अपने चरित्र और व्यक्तित्व को बेहतर बनाने के लिए पाठ्यक्रम (सामाजिक और व्यावसायिक नैतिकता) सीखने के दौरान क्या उपाय किए हैं?
2. स्वामी विवेकानन्द की जीवन यात्रा के सन्दर्भ में आध्यात्मिकता की अवधारणा पर चर्चा करें।
3. पंडित मदन मोहन मालवीय के जीवन रेखा पर एक लघु निबंध लिखिए।
4. भ्रष्टाचार से आप क्या समझते हैं? क्या यह एक सामाजिक समस्या है या व्यक्तिगत समस्या? भ्रष्टाचार की समस्या को रोकने के लिए उपचारात्मक उपाय सुझाएँ।



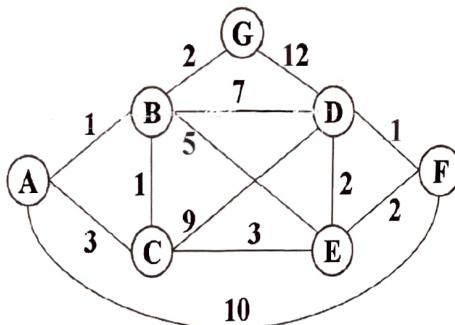
NATIONAL INSTITUTE OF TECHNOLOGY PATNA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
END SEMESTER EXAMINATION – DECEMBER 2023
B. Tech (Computer Science & Engineering) 3rdSemester (SECTION-A,B,& C)
Time: 3 Hrs. CS34104– Data Structures Max.Marks:60

Instruction:

1. Answer all questions.
2. All parts of the question must be answered at the same place.
3. The Marks, CO (Course Outcome) and BL (Bloom's Level) related to questions are mentioned on the right-hand side margin.U(Understand), R(Remember), A(Apply) , E(Evaluate) , C(Create), An(Analyze).

Q.No	Question	Marks/ CO and BL
1	<p>a) Let us assume A is the array to be sorted. Consider the two equal elements X, Y positioned at index i and j in the unsorted array A, where $i < j$; after sorting X is at position p and Y is at position q in the sorted array A with the property $p < q$; if any sorting algorithm follows this property for all equal elements in A then it is called stable sorting (i.e. their relative order is maintained in both unsorted array and in sorted array). Which of the following sorting algorithm(s) are stable sorting algorithms and which are not stable sorting algorithms?</p> <p>i) Bubble sort ii) Heap sort iii) Merge sort iv) Quick sort</p> <p>b) Execute Merge sort algorithm on the following data till all key values are placed in their position 12, 34, 45, 15, 4, 11, 7, 8 5, 14, 35, 89, 43, 21, 6.</p>	4 M CO4 U 6 M CO4 A
2	<p>a) Consider the in-order and pre-order traversals of a binary search tree are (1, 2, 3, 4, 5, 6, 8, 10, 25) and (4, 3, 1, 2, 10, 8, 5, 6, 25) respectively. Construct the unique binary search tree for the given in-order and pre-order traversals.</p> <p>b) Write the recursive functions for in-order, pre-order, and post-order traversals for a binary search tree.</p>	5M CO3 C 5M CO3 An
3	<p>a) Show the AVL tree that results after each of the integer keys 10, 28, 51, 16, 3, 22, and 37 are inserted, in that order, into an initially empty AVL tree and also mention any rotations used in this process.</p> <p>b) Write the worst-case time complexities for the following operations:</p> <ol style="list-style-type: none">i) Searching element in a Binary search tree.ii) Traversing the graph using Depth First Search.iii) Sorting elements using Insertion sort.iv) Inserting an element into AVL tree.	6M CO3 A 4M CO4 R

4	<p>a) Distinguish between Prims algorithm and Kruskal algorithms. Execute both prims and Kruskal algorithms on the following undirected weighted graph and show the result after each step.</p> <p>$G = (\{A, B, C, D, E, F, G, H\}, \{(A, B, 3), (A, D, 4), (B, D, 5), (B, C, 1), (C, F, 8), (C, H, 7), (D, E, 2), (F, H, 3), (H, G, 1), (D, F, 2), (A, E, 9)\})$.</p>	10M CO3 A
5	<p>Define B-Tree and how it differs from Binary Search Tree (BST). Is BST being also a B-Tree? If so, state the reason. Otherwise, give an example of BST, which is not a B-Tree. Give the internal node structures of the B-Tree of orders four and five. Construct the B-Tree of order three on an initially empty B-Tree using the below given data in the same order (You must show the resultant tree after inserting each element in the tree).</p> <p style="text-align: center;">3, 1, 4, 5, 12, 2, 6, 8, 7, 9, 11, 23.</p> <p>Note: Follow the following constraints when constructing the B-Tree:</p> <ul style="list-style-type: none"> i. The root is either a leaf or has between 2 and M children. ii. All non-leaf nodes except the root node have between $\text{ceil}(M/2)$ to M children. iii. The number of keys in a (non-root) leaf is also between $\text{ceil}(M/2)$ and M. 	10M CO3 U&E
6	<p>a. Write the functions for en-queue and de-queue operations of Queue using singly linked list.</p> <p>b. Apply Dijkstra's algorithm on the following graph. You must show all the process steps and draw the final shortest graph. Consider vertex A as the source vertex.</p>	5M CO2 R 5M CO3 U





[Attempt all questions; Answer concisely only in blue/black ink; Use pencil for artwork;

Assume missing data; No mobile phones]

[Marks, Course Outcome and Bloom's Level are mentioned on right-hand side of each question]

Sl	Questions (Symbols have their usual meaning)	CO	BL
1.	a. For the following tree, compute its diameter, radius and center. (7) Fig: Tree, with vertex set $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$. उपरोक्त ट्री के लिए, उसके डायमेटर (व्यास), रॉडियस (त्रिज्या) और सेंटर (केंद्र) की गणना करो। [Course outcome(s) evaluated: CO-4(Apply/Solve)]	CO-4	Level-3
	b. Prove that one and only one path (i.e., a unique path) is present between every pair of vertices in any undirected tree. (5) साबित करो कि किसी भी अँनडिरेक्टेड (अनिर्देशित) ट्री में वर्टीसस (शीर्षों) की प्रत्येक जोड़े के बीच एक और केवल एक पथ (अर्थात्, एक अद्वितीय पथ) मौजूद है। [Course outcome(s) evaluated: CO-6(Analyze/Prove)]	CO-6	Level-4
2.	a. Represent the following tree in the form of Prüfer Sequence. Elaborate all intermediate steps involved in the representation. (10) Fig: Tree, with vertex set $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$. उपरोक्त ट्री को पुफ़र सीकेस (अनुक्रम) के रूप में रिप्रेज़ेट करो। उस रिप्रेज़ेशन प्रस्तुत करने में शामिल सभी इंटरमेडिएट (मध्यवर्ती) स्टेप्स का विस्तार से वर्णन करो। [Course outcome(s) evaluated: CO-3(Understand/Represent)]	CO-3	Level-2
	b. How many labeled trees of n vertices are possible? (2) n वर्टीसस (शीर्षों) के कितने लेबल वाले ट्री संभव हैं? [Course outcome(s) evaluated: CO-1(Remember/Recall)]	CO-1	Level-1

SI	Questions (Symbols have their usual meaning)	CO BL
3.	a. For the following data center network of a certain enterprise, use the properties and theorems of connected graph to identify — (i) how many vertices are to be removed at the least to disconnect the network, and (ii) how many edges are to be removed at the least to disconnect the network? Show intermediate steps. (10)	
		CO-5 Level-3
	Fig: Some data center network spanning multiple cities.	
	किसी भी एक इंटरप्राइज के उपरोक्त डेटा सेंटर नेटवर्क के लिए, निर्धारित करों — (i) उस नेटवर्क को डिस्कनेक्ट करने के लिए कम से कम कितने वर्टेस्स (शीर्षों) को हटाए जाने हैं, और (ii) उस नेटवर्क को डिस्कनेक्ट करने के लिए कम से कम कितने एजेस (किनारों) को हटाया जाना है? वह निर्धारित करने के लिए कनेक्टेड ग्राफ के प्रॉपर्टीज (गुणों) और थिओरम्ज (प्रमेयों) का उपयोग करो। इंटरमेडिएट (मध्यवर्ती) स्टेप्स दिखाएँ।	
	[Course outcome(s) evaluated: CO-5(Apply/Use)]	
b.	Give any two equivalent definitions of tree in graph theory. (2) ग्राफ थिओरी (सिद्धांत) में ट्री की कोई दो इकुईवॉलेट (समतुल्य) परिभाषाएँ दीजिए।	CO-1 Level-1
	[Course outcome(s) evaluated: CO-1(Remember/Recall)]	
4.	a. Prove that in any undirected graph, the sum of the degrees of all its vertices is same as twice the number of edges of the graph. (5) साबित करों कि किसी भी अॉनडिरेक्टेड (अनिर्देशित) ग्राफ में, उसके सभी वर्टेस्स (शीर्षों) की डिग्री का योग उस ग्राफ के एजेस (किनारों) की कुल संख्या के दोगुने कि समान होता है।	CO-6 Level-4
	[Course outcome(s) evaluated: CO-6(Analyze/Prove)]	
b.	Use the properties and theorems of graphic sequence of simple graphs to test whether the given degree sequence $\{d_n\}: 4, 3, 3, 3, 3$ (where $n > 1, n \in \mathbb{Z}^+$) is graphic or not. Show intermediate steps. (7) जांचो कि दिया गया डिग्री सीकेस (अनुक्रम) $\{d_n\}: 4, 3, 3, 3, 3$ (जहाँ $n > 1, n \in \mathbb{Z}^+$) ग्राफिक हैं या नहीं। वह जांचने के लिए सरल ग्राफ के ग्राफिक सीकेस (अनुक्रम) के प्रॉपर्टीज (गुणों) और थिओरम्ज (प्रमेयों) का उपयोग करो। इंटरमेडिएट (मध्यवर्ती) स्टेप्स दिखाएँ।	CO-5 Level-3
	[Course outcome(s) evaluated: CO-5(Apply/Use)]	
5.	a. Consider the Tower of Hanoi puzzle with 3 pegs and n disks. Also consider that the recurrence relation $H_n = 2H_{n-1} + 1$ denote the number of moves needed to solve the puzzle, with initial condition $H_1 = 1$. Use the closed formula of recursively defined sequence $\{H_n\}$ to determine the closed-form solution of recurrence relation of given puzzle. (8)	CO-5 Level-3

SI	Questions (Symbols have their usual meaning)	CO BL
5. a.	<p>मान लों टावर ऑफ हनोई पहेली हैं, जिसमें 3 खूटियाँ और n डिस्क हैं। यह भी मान लों कि रेकरेस (पुनरावृत्ति) संबंध $H_n = 2H_{n-1} + 1$, जिसका प्रारंभिक स्थिति $H_1 = 1$ है, वह उस पहेली को हल करने के लिए आवश्यक मूवस (चालों) की संख्या को दर्शाता है। दी गई पहेली के रेकरेस संबंध का क्लोज़्ड-फॉर्म (बंद-रूप) समाधान निर्धारित करो। वह निर्धारित करने के लिए रेकर्सिवली डिफाइंड सीकेंस $\{H_n\}$ के क्लोज़्ड फॉर्मूला (बंद सूत्र) का उपयोग करो।</p> <p>[Course outcome(s) evaluated: CO-5(Apply/Use)]</p>	CO-5 Level-2
b.	<p>Define reflexivity, symmetry, antisymmetry and transitivity of binary relation. (4)</p> <p>बाइनरी रिलेशन (द्विआधारी संबंध) की रिफ्लेक्सिविटी (प्रतिवर्तीता), सिमेट्री (समरूपता), एंटीसिमेट्री (प्रतिसममिति) और ट्रूंसटिविटी (परिवर्तनशीलता) की परिभाषा लिखों।</p> <p>[Course outcome(s) evaluated: CO-1(Remember/Recall)]</p>	CO-1 Level-1

List of Course Outcomes of "Discrete Mathematics and Graph Theory": After completing this course, a student should be able to —

- CO-1. *recall* mathematical logic terminologies and principles, discrete structures terminologies and principles, counting principles, recurrence relation terminologies and principles, graph terminologies and principles, tree terminologies and principles, as well as graph coloring and covering principles;
 [Bloom level: Remember; Mapped to: PO-1]
- CO-2. *explain* concepts of logical equivalence, quantification, sum rule, product rule, inclusion-exclusion, pigeonhole, generating functions, graph connectedness, Euler trails and Hamilton paths, shortest path rule, tree traversal, spanning tree, graph planarity, coloring rule, covering and portioning of graph;
 [Bloom level: Understand; Mapped to: PO-1, PO-2]
- CO-3. *represent* given instructions on contemporary scenarios through graphs and trees;
 [Bloom level: Understand; Mapped to: PO-1, PO-2]
- CO-4. *solve* problems on propositional and predicate logic, set properties and operations, relation properties and operations, function properties and operations, sequence properties and operations, product and sum rules, inclusion-exclusion principle, pigeonhole principle, permutations and combinations, recurrence relation properties, generating functions, graph isomorphism, graph traversal properties and principles, tree traversal properties and operations, planar graph properties and rules, coloring rules of graphs, graph covering and partitioning;
 [Bloom level: Apply; Mapped to: PO-1, PO-2, PO-3]
- CO-5. *use* graph and tree models, as well as planar representations and chromatic numbering of graphs to fulfil given objectives on contemporary network infrastructures;
 [Bloom level: Apply; Mapped to: PO-1, PO-2, PO-3]
- CO-6. *prove* by constructing argument form to establish given properties of specified discrete structures.
 [Bloom level: Analyze; Mapped to: PO-3, PO-5]

----- "BEST OF LUCK" -----



NATIONAL INSTITUTE OF TECHNOLOGY PATNA
END- SEMESTER EXAMINATION, DEC 2023

Program: B.Tech. (CSE)

Semester: 3rd

Department: CSE

Course Code: CS34111

Full Marks: 60

Course Name: Operating Systems

Duration of Examination: 3 hours

INSTRUCTIONS

- Assume Missing data if any
- Answer all the question
- SECTION I is common to CSE-I, CSE-II and CSE-III students
- SECTION II is to be attempted by CSE-I and CSE-II Students
- SECTION III is to be attempted only by CSE-III students
- Students are requested to write the Section they belong on top of the first page of the answer script.

SECTION I

S.No.	QUESTION	CO	BL	Marks																											
1.	<p>Consider the set of 4 processes whose arrival time and burst time are given below-</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th rowspan="2">Process No.</th> <th rowspan="2">Arrival Time</th> <th rowspan="2">Priority</th> <th colspan="3">Burst Time</th> </tr> <tr> <th>CPU Burst</th> <th>I/O Burst</th> <th>CPU Burst</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>0</td> <td>2</td> <td>1</td> <td>5</td> <td>3</td> </tr> <tr> <td>P2</td> <td>2</td> <td>3</td> <td>3</td> <td>3</td> <td>1</td> </tr> <tr> <td>P3</td> <td>3</td> <td>1</td> <td>2</td> <td>3</td> <td>1</td> </tr> </tbody> </table> <p>If the CPU scheduling policy is Priority Scheduling. Draw the Gantt chart and calculate the average waiting time and average turn around time. (Lower number means higher priority)</p>	Process No.	Arrival Time	Priority	Burst Time			CPU Burst	I/O Burst	CPU Burst	P1	0	2	1	5	3	P2	2	3	3	3	1	P3	3	1	2	3	1	CO-1	2	4+3+3
Process No.	Arrival Time				Priority	Burst Time																									
		CPU Burst	I/O Burst	CPU Burst																											
P1	0	2	1	5	3																										
P2	2	3	3	3	1																										
P3	3	1	2	3	1																										
2.	<p>A system uses 4-page frames for storing process pages in main memory. It uses the Least Recently used (LRU) and Most Recent Used (MRU) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string - 7,0,1,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1. Also calculate the hit ratio and miss ratio.</p>	CO-3	3	4+3+3																											
3.	<p>Consider a single level paging scheme. The virtual address space is 256 MB and page table entry size is 4 bytes. What is the minimum page size possible such that the entire page table fits well in one page?</p>	CO-3	4	10																											

<p>4. An operating system uses the banker's algorithm for deadlock avoidance when managing the allocation of three resource types X, Y and Z to three processes P0, P1 and P2. The table given below presents the current system state. Here, the Allocation matrix shows the current number of resources of each type allocated to each process and the Max matrix shows the maximum number of resources of each type required by each process during its execution.</p> <table border="1" style="margin-top: 10px; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="3">Allocation</th><th colspan="3">Max</th></tr> <tr> <th rowspan="2"></th><th>X</th><th>Y</th><th>Z</th><th>X</th><th>Y</th><th>Z</th></tr> </thead> <tbody> <tr> <td>P0</td><td>0</td><td>0</td><td>1</td><td>8</td><td>4</td><td>3</td></tr> <tr> <td>P1</td><td>3</td><td>2</td><td>0</td><td>6</td><td>2</td><td>0</td></tr> <tr> <td>P2</td><td>2</td><td>1</td><td>1</td><td>3</td><td>3</td><td>3</td></tr> </tbody> </table> <p>There are 3 units of type X, 2 units of type Y and 2 units of type Z still available. The system is currently in safe state. Consider the following independent requests for additional resources in the current state-</p> <p>REQ1: P0 requests 0 units of X, 0 units of Y and 2 units of Z REQ2: P1 requests 2 units of X, 0 units of Y and 0 units of Z</p> <p>Which of the following is TRUE?</p> <ul style="list-style-type: none"> A. Only REQ1 can be permitted B. Only REQ2 can be permitted C. Both REQ1 and REQ2 can be permitted D. Neither REQ1 nor REQ2 can be permitted 	Allocation			Max				X	Y	Z	X	Y	Z	P0	0	0	1	8	4	3	P1	3	2	0	6	2	0	P2	2	1	1	3	3	3	CO-2 3 10
Allocation			Max																																
	X	Y	Z	X	Y	Z																													
	P0	0	0	1	8	4	3																												
P1	3	2	0	6	2	0																													
P2	2	1	1	3	3	3																													

SECTION II

S.No.	QUESTION	CO	BL	Marks		
5.	<p>Consider a system using multilevel paging scheme. The page size is 1 GB. The memory is byte addressable and virtual address is 72 bits long. The page table entry size is 4 bytes. Find-</p> <ul style="list-style-type: none"> a. How many levels of page table will be required? b. Give the divided physical address and virtual address. 	CO-3	3	10		
6.	<p>Suppose we want to synchronize two concurrent processes P and Q using binary semaphores S and T. The code for the processes P and Q is shown below-</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="vertical-align: top; width: 50%;"> Process P: <pre> while (1) { W: print '0'; print '0'; X: } </pre> </td> <td style="vertical-align: top; width: 50%;"> Process Q: <pre> while (1) { Y: print '1'; print '1'; Z: } </pre> </td> </tr> </table> <p>Synchronization statements can be inserted only at points W, X, Y and Z. Which of the following will always lead to an output string with '001100110011'?</p> <ul style="list-style-type: none"> A. P(S) at W, V(S) at X, P(T) at Y, V(T) at Z, S and T initially 1 B. P(S) at W; V(T) at X, P(T) at Y, V(S) at Z, S initially 1 and T initially 0 C. P(S) at W, V(T) at X, P(T) at Y, V(S) at Z, S and T initially 1 D. P(S) at W, V(S) at X, P(T) at Y, V(T) at Z, S initially 1 and T initially 0 	Process P: <pre> while (1) { W: print '0'; print '0'; X: } </pre>	Process Q: <pre> while (1) { Y: print '1'; print '1'; Z: } </pre>	CO-2	4	10
Process P: <pre> while (1) { W: print '0'; print '0'; X: } </pre>	Process Q: <pre> while (1) { Y: print '1'; print '1'; Z: } </pre>					

SECTION III

S.No.	QUESTION	CO	BL	Marks
5.	Consider a paging hardware with a TLB. Assume that the entire page table and all the pages are in the physical memory. It takes 10 milliseconds to search the TLB and 80 milliseconds to access the physical memory. If the TLB hit ratio is 0.6, then what is the effective memory access time?	CO-3	2	10
6.	Suppose a disk has 201 cylinders, numbered from 0 to 200. At some time the disk arm is at cylinder 100, and there is a queue of disk access requests for cylinders 30, 85, 90, 100, 105, 110, 135, and 145. If Shortest-Seek Time First (SSTF) is being used for scheduling the disk access, the request for cylinder 90 is serviced after servicing _____ the number of requests.	CO-4	3	10

***** END *****



NATIONAL INSTITUTE OF TECHNOLOGY PATNA
Department of Computer Science & Engineering
End Semester Examination, December 2023

BTech: 3rd Semester (CSE-1, CSE-2, CSE-3, CSDD, ECE)

Course Name: Database Management Systems

Course Code: CS34109

Maximum Time: 3 hours

Max. Marks: 60

Note: 1. Answer all questions & sub-questions must be answered sequentially in one place.
2. Write the answers neatly.

Q.No	Question	Marks	CO	BL
1	<p>Consider the following relation schema for the BANK database:</p> <p>BRANCH (<u>BranchID</u>, Bname, City, Phone)</p> <p>ACCOUNT (<u>AccountNo</u>, Aname, AType, <u>BranchID</u>, Balance)</p> <p>TRANSACTION (<u>TID</u>, T_Date, T_Type, <u>AccountNo</u>, Amount)</p> <p>Based on the above relational schema, write the following queries in SQL.</p> <p>i. Retrieve the ID and name of all the branches located in <i>Patna</i> city.</p> <p>ii. Retrieve the ID, type, and amount of all the transactions of withdrawal type.</p> <p>iii. List the number and type of all accounts opened in the branch having ID <i>B010</i>.</p> <p>iv. List the number and name of account holders withdrawing an amount greater than 10,000 on 31st March 2007.</p> <p>v. List the number and name of account holders having savings account in the city of <i>Patna</i>.</p>	15	CO4	3
2 a)	Three constraints may apply to a specialization/generalization. What are they? Explain each of them with a suitable example.	7	CO2	1
b)	<p>The hospital management wants to build the database for a newly constructed hospital. They explain the database scenario for a hospital to the database designer in the following manner: Hospital staff consist of doctors and nurses. Each hospital staff has a sid, name (first name, middle name, and last name), and mobile numbers. Doctors have an office and up to 4 specializations. Doctors have nurses who work with/assist them. Each nurse has one specialization and belongs to a specific department in the hospital. All nurses work with doctors (no exception). Many nurses can work with one doctor. Doctors sometimes consult with a colleague (who is another doctor). Doctors treat patients. Patient information collected by the hospital consists of pid, name, insurance, date of admission, and checked-out date. A patient can undergo several tests. Doctors perform tests. Each test has a unique ID, name, result, and date and time the test was performed.</p> <p>i) Create an E-R diagram for the above database scenario. Make sure to include an indication of the cardinality of relationships and specify any mandatory relationship (total participation). State any assumptions that you make.</p> <p>ii) Convert the E-R diagram to tables using schema statements.</p>	8	CO2	2,3

3	a)	Consider a relation schema R (A, B, C, D, E, G) with a set of functional dependencies F {A→B, BC→D, D→E, D→G}. Find the closure of F.	7	CO6	3														
	b)	Sup (Sno, Sname, City, Status, Pno), Proj (Pno, Qty), Sno → {Sname, City} City → {Status}, {Sno, Pno} → Qty Convert it up to 3NF.	8	CO6	3														
4	a)	Test the below schedule whether it is a conflict serializable schedule or not. If the schedule is serializable, write the equivalent serializable schedule.	8	CO8	3														
	b)	<p style="text-align: center;">Schedule 1</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>T1</th> <th>T2</th> </tr> </thead> <tbody> <tr> <td>Read (X)</td> <td></td> </tr> <tr> <td>Read (Y)</td> <td></td> </tr> <tr> <td></td> <td>Read (X)</td> </tr> <tr> <td></td> <td>Read (Y)</td> </tr> <tr> <td></td> <td>Write (Y)</td> </tr> <tr> <td>Write (X)</td> <td></td> </tr> </tbody> </table> <p>Construct the final reduced relational algebra tree of the following SQL query:</p> <pre> SELECT p.propertyNo, p.street FROM Client c, Viewing v, PropertForRent p WHERE c.prefType=Flat AND c.clientNo=v.clientNo AND v.propertyNo=p.propertyNo AND c.maxRent>=p.rent AND c.prefType=p.type AND p.ownerNo=CO93; </pre>	T1	T2	Read (X)		Read (Y)			Read (X)		Read (Y)		Write (Y)	Write (X)		7	CO7	3
T1	T2																		
Read (X)																			
Read (Y)																			
	Read (X)																		
	Read (Y)																		
	Write (Y)																		
Write (X)																			



राष्ट्रीय प्रौद्योगिकी संस्थान पटना

End Semester Exam (Jul-Dec'23)

National Institute of Technology Patna

Session: 2023-24 Autumn'23 Semester

Department: Computer Science and Engineering

Programme: BTech(CSE-I,II,III), DD(CSE-CS,DS)

Semester: 3rd

Course Code: **CS34105**

Course: Object Oriented Programming

Full Marks: 60

Duration: 3 hours

[Attempt all questions; Answer concisely only in blue/black ink; Use pencil for artwork;
Assume missing data; No mobile phones]

[Marks, Course Outcome and Bloom's Level are mentioned on right-hand side of each question]

Sl.	Questions	CO	BL
1. a.	<p>Write a Java program that creates a class 'Member', which has instance variables 'income' of type int and 'name' of type String. Write a method 'int AddIncome(Member m1, Member m2)' that adds the income of two family members. Write another class 'Demo' having 'main()' method, and create three objects of 'Member' inside the 'Demo' class. Further, add the income of two 'Member' objects using 'AddIncome' method, and make it equal to the income of the third 'Member' object. Serialize the third 'Member' object, and afterward deserialize it and display the income of third 'Member' object. (12)</p> <p>एक क्लॉस 'मेंबर' ('Member') बनाने के लिए एक Java प्रोग्राम लिखिए, जिसमें इंट (int) टाइप के 'इनकम' ('income') और स्ट्रिंग (String) टाइप के 'नेम' ('name') जैसे इंस्टेंस वेरिएबल्स हों। रिटर्न टाइप इंट (int) की एक मेथड 'AddIncome (Member m1, Member m2)' लिखिए, जिससे परिवार के दो सदस्यों ('मेंबर्स') की आय जुड़ जाती हैं। एक और 'डेमो' ('Demo') क्लॉस लिखिए, जिसमें 'main()' मेथड रखना है, और 'डेमो' क्लॉस के अंदर 'मेंबर' ('Member') की तीन ऑब्जेक्ट्स बनाईं। फिर, 'AddIncome' मेथड का उपयोग करके दो 'मेंबर' ('Member') ऑब्जेक्ट्स की आय जुड़िए, और इसे तीसरे 'मेंबर' ('Member') ऑब्जेक्ट की आय के बराबर कीजिए। इस तीसरे 'मेंबर' ('Member') ऑब्जेक्ट को सेरिएलाइज़ कीजिए, और बाद में इसे डिसेरिएलाइज़ तथा तीसरे 'मेंबर' ('Member') ऑब्जेक्ट की आय का मान प्रदर्शित कीजिए।</p> <p>[Course outcome(s) evaluated: CO-5(Apply/Use)]</p>	CO-5	Level-3
b.	<p>Write down the characteristics of object oriented programming. (3)</p> <p>ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग की विशेषताएँ लिखिए।</p> <p>[Course outcome(s) evaluated: CO-1(Remember/Recall)]</p>	CO-1	Level-1
2. a.	<p>Write a Java program that implements the 'IQueue' interface given in the next page. Your program must incorporate the following. (10)</p> <ul style="list-style-type: none"> Create class 'MyQueue' implementing 'IQueue' interface, wherein add appropriate fields, public constructor(s), and overriding 'public String toString()' method. This class must be accessible and usable from any package, and must have proper encapsulation. However, any attempt to extend this class should produce a compile-time error. 	CO-5	Level-3

Sl.
2. a.

Questions

```
public interface IQueue {  
    public int size(); //Returns number of objects in queue  
    public boolean isEmpty(); //Returns true if queue is empty  
    public void enqueue(Object o); //Adds an item to tail of queue  
    public Object dequeue(); //Removes and returns item from  
                           // head of queue  
}
```

- A sample demo of main() method using 'MyQueue' is shown below.

```
public static void main(String[] args) {  
    MyQueue lst = new MyQueue();  
    lst.enqueue("Hello");  
    lst.enqueue("World");  
    System.out.println(lst.dequeue());  
    System.out.println(lst.dequeue());  
}
```

Output::

Hello
World

एक Java प्रोग्राम लिखिए जो उपरोक्त 'IQueue' इंटरफ़ेस को इम्पलिमेंट करता है। उस प्रोग्राम में निम्नलिखित शामिल होना चाहिए।

- क्लॉस 'MyQueue' बनाएं, जहाँ 'IQueue' इंटरफ़ेस इम्पलिमेंट हों। उस क्लॉस में उपयुक्त फ्रील्ड, public कंस्ट्रक्टर्स जोड़ें, और 'public String toString()' मेथड को ओवरराइड करें। यह क्लॉस किसी भी पैकेज से पहुंच योग्य और प्रयोग करने योग्य होनी चाहिए और इसमें उचित इनकैप्सुलेशन होना चाहिए। हालाँकि, इस क्लॉस को एक्सटेंड करने के किसी भी प्रयास से कंपाइल-समय एर्रर उत्पन्न होनी चाहिए।
- 'MyQueue' का उपयोग करके main() विधि का एक नमूना डेमो ऊपर दिखाया गया है।

[Course outcome(s) evaluated: CO-5(Apply/Use)]

- b. Explain the outcome, in the form of either listing of error(s) (if any) or the output, of the following Java program. Consider they are saved in 'Boat.java', 'Rowboat.java', 'TestBoats.java' and 'Sailboat.java' within the same directory 'exmpgrgs'. (5)

```
package exmpgrgs;  
public class Boat {  
    private int length ;  
    public void setLength ( int len ) {  
        length = len;  
    }  
    public int getLength() {  
        return length ;  
    }  
    public void move() {  
        System.out.print("drift ");  
    }  
}
```

```
package exmpgrgs;  
public class Rowboat extends Boat {  
    public void rowTheBoat() {  
        System.out.print("stroke ");  
    }  
}
```

CO-3
Level-3

CO-5
Level-3

Questions

CO-BL

SI. 2. b.

```

package exmpargs;
public class Sailboat extends Boat {
    public void move() {
        System.out.print("hoist sail ");
    }
}

package exmpargs;
public class TestBoats {
    public static void main(String[] args) {
        Boat b1 = new Boat();
        Sailboat b2 = new Sailboat();
        Rowboat b3 = new Rowboat();
        b2.setLength(32);
        b1.move();
        b3.move();
        b2.move();
    }
}

```

CO-3
Level-2

ऊपर दिया गया Java प्रोग्रेम की परिणाम को व्याख्या कीजिए। परिणाम एरोर(एरोर्स) की सूची (यदि कोई हो) या आउटपुट के रूप में लिखिए। मान लें कि प्रोग्रेम 'Boat.java', 'Rowboat.java', 'Sailboat.java' और 'TestBoats.java' में एक ही डायरेक्टरी 'exmpargs' में सेव हैं।

[Course outcome evaluated: CO-3(Understand/Explain)]

- 3. a.** Write a Java program to create your own Exception, named 'Traffic' exception. Write a class 'Demo' having 'main()' method that will take String input from the user, and if the user has entered String having more than five characters, then it will throw 'Traffic' exception. (6)

एक Java प्रोग्रेम लिखिए, जिसमें अपना खुद का एक्सेप्शन बनाएं, जिसका नाम 'ट्रैफिक' एक्सेप्शन हो। एक 'डेमो' ('Demo') क्लॉस लिखिए, जिसमें 'main()' मेथड रखना है, और 'डेमो' क्लॉस यूज़र (उपयोगकर्ता) से स्ट्रिंग इनपुट लेगा। और यदि यूज़र ने पाँच से अधिक वर्ण वाली स्ट्रिंग दर्ज की हैं, तो यह 'ट्रैफिक' एक्सेप्शन थ्रो देगा।

[Course outcome(s) evaluated: CO-5(Apply/Use)]

CO-5
Level-3

- b.** Identify the differences between the keywords 'throw' and 'throws' in Java. (3)

Java में कौवड 'थ्रो' ('throw') और 'थ्रोज़' ('throws') के बीच अंतर लिखिए।

[Course outcome(s) evaluated: CO-2(Remember/Recognise)]

CO-2
Level-1

- c.** Explain the outcome, in the form of either listing of error(s) (if any) or the output, of the following Java program. Consider it is saved in 'MonsterTestDrive.java' within the directory 'exmpargs'. (3)

```

package exmpargs;
public class MonsterTestDrive {
    public static void main(String[] args) {
        Monster[] monsters = new Monster[3];
        monsters[0] = new Vampire();
        monsters[1] = new Dragon();
        monsters[2] = new Monster();
        for (int i = 0; i < monsters.length; i++) {
            monsters[i].frighten(i);
        }
    }
}

```

CO-3
Level-2

Sl.

3. c.

Questions

```

class Monster {
    boolean frighten(int x) {
        System.out.println("arrgh");
        return false;
    }
}

class Vampire extends Monster {
    boolean scare(int x) {
        System.out.println("a bite?");
        return true;
    }
}

class Dragon extends Monster {
    boolean frighten(int degree) {
        System.out.println("breathe fire");
        return true;
    }
}

```

ऊपर दिया गया Java प्रोग्रेम की परिणाम को व्याख्या कीजिए। परिणाम एरर(एरर्स) की सूची (यदि कोई हो) या आउटपुट के रूप में लिखिए। मान लों कि प्रोग्रेम 'MonsterTestDrive.java' में 'exmpgns' डायरेक्टरी में सेव हैं।

[Course outcome evaluated: CO-3(Understand/Explain)]

d. Distinguish between abstract class and interface in Java.

(3)

Java में एक्सट्रैक्ट क्लॉस और इंटरफ़ेस के बीच अंतर लिखिए।

[Course outcome(s) evaluated: CO-2(Remember/Recognise)]

4. a. Consider the following program.

```

class Employee {
    protected String name;
    public Employee() { this.name = ""; }
    public Employee(String nm) {
        this.name = "Employee: " + nm;
    }
    public String getName() { return this.name; }
} //end of class

class Manager extends Employee {
    private String name;
    public Manager(String nm) {
        this.name = "Manager: " + nm;
    }
    public String getName() { return this.name; }
} //end of class

public class CstDemo {
    public static void printName(Object ob) {
        String name = ob.getName();
        System.out.println(name);
    }
} //end of printName()

public static void main(String args[]) {
    Employee emp;
    Manager mgr;
    emp = new Manager("Abcd");
}

```

CO-BL

CO-3
Level-2CO-2
Level-1CO-4
Level-3

• Sl.
4. a.

Questions

COBL

```
    printName(emp);
    mgr = emp;
    printName(mgr);
}//end of main()
}//end of class
```

Find the compilation errors and debug them, if any. (5)

ऊपर दिया गया प्रोग्रैम पर गौर करें। वह प्रोग्रैम की कंपाइलेशन एर्रर्स का पता लगाएं, और यदि कोई हो तो उन्हें डीबग करें।

[Course outcome(s) evaluated: CO-4(Apply/Solve)]

b. Identify the purpose of the keyword 'this' in Java. (2)

Java में कौवड़ 'दिस' ('this') का उद्देश्य लिखिए।

[Course outcome(s) evaluated: CO-2(Remember/Recognise)]

c. Check the following program for any error(s), and if so, what is/are the error(s) and write the correct program along with output. (8)

```
class A {
    final int ivar = 7;
    void m1() {
        System.out.print("A's m1, ");
    }
    void m2() {
        System.out.print("A's m2, ");
    }
    final void m3() {
        System.out.print("A's m3, ");
    }
}
class B extends A {
    void m1() {
        System.out.print("B's m1, ");
    }
}
class C extends B {
    void m3() {
        System.out.print("C's m3, " + (ivar + 6));
    }
}
public class Inherit{
    public static void main(String[] args) {
        A a = new A();
        B b = new B();
        C c = new C();
        A a2 = new C();
        // Block 1
        {
            a.ivar = 8;
            b.m1();
            c.m2();
        }
    }
}
```

CO-1

Level-3

CO-2

Level-1

CO-4
Level-3

```

        a.m3();
    }
    // Block 2
    {
        a2.m1();
        a2.m2();
        a2.m3();
    }
    // Block 3
    {
        c.m1();
        c.m2();
        c.m3();
    }
}

```

जाँचिए कि उपरोक्त प्रोग्रेम में किसी भी एरोर(एरोस) हैं या नहीं। और यदि हैं, तो एरोर(एरोस) क्या हैं लिखिए, तथा आउटपुट के साथ सही प्रोग्रेम भी लिखिए।

[Course outcome(s) evaluated: CO-4(Apply/Solve)]

List of Course Outcomes of "Object Oriented Programming": After completing this course, a student should be able to —

CO-1. *recall* object oriented programming terminologies and principles, as well as fundamental concepts and programming practices of object oriented programming platforms;

[Bloom level: Remember; Mapped to: PO-1]

CO-2. *recognise* object oriented programming structures for building multidimensional data arrays, inherited and polymorphic data structures, exception handlers, input/output handlers, multithreading;

[Bloom level: Remember; Mapped to: PO-1]

CO-3. *explain* concepts of object orientation, arrays, control blocks, looping, nesting, access modifiers, overloading, inheritance, polymorphism, overriding, dynamic method lookup and dispatch, abstract classes, interfaces, packages, checked and unchecked exceptions, chained exceptions, input/output streams, multithreading;

[Bloom level: Understand; Mapped to: PO-1, PO-2]

CO-4. *solve* problems on selections, iterations, arrays, objects, classes, fields, methods, access modifiers, object passing and return, overloading, boxing, wrapper class, objects of arrays, string handling, overriding, inheritance hierarchy, abstract class, upcasting, downcasting, association, composition, compile-time polymorphism, runtime polymorphism, dynamic method dispatch, interface implementation, class nesting, try-catch, throwing, rethrowing, byte streams, character streams, multithreading;

[Bloom level: Apply; Mapped to: PO-1, PO-2, PO-3]

CO-5. *use* methodologies of object oriented programming in writing programs;

[Bloom level: Apply; Mapped to: PO-1, PO-2, PO-3]

CO-6. *implement* programs in hands-on experiments to fulfil given problem objectives.

[Bloom level: Apply; Mapped to: PO-3, PO-5]

----- "BEST OF LUCK" -----