Course Plan

Course Handout

Institute/School Name	Chitkara University Institute of Engineering & Technology		
Department Name	Department of Computer Science and Engineering		
Programme Name	B.E (Computer Science & Technology)		
Course Name	Data Structures using Object Oriented Programming	Session	July – Dec 2025
Course Code	24CSE0208	Semester/Batch	3rd /2024
L-T-P(Per Week)	2-0-8	Course Credits	06
Pre-requisite	Basic concepts of computer fundamentals	NHEQF Level[1]	05[1]
Course Coordinator	Dr. Heena Wadhwa	SDG Number[4]	4, 8, 9[4]

1. Objectives of the Course

This course is an introduction to data structures using the concept of object oriented programming. It is to provide students with practical ability in the Java language, which can be used for organizing and managing data for software development. The main objectives are:

- To apply core object-oriented programming principles to design and implement algorithms.
- To apply control structures to control the flow of program execution, facilitating efficient problem-solving and decision-making
- To understand and implement fundamental data structures such as stacks, queues and linked lists for effective problem-solving
- To evaluate time-space complexity trade-offs in algorithm design and implement efficient searching and sorting techniques using data structures

2. Course Learning Outcomes (CLOs)

Student should be able to:

	CLOs	Program Outcomes (PO)	NHEQF Level Descriptor[2]	No. of Lectures
CLO01	Understand the principles of object oriented programming and implement basic programming for logic building by using data types, variables, operators and expressions	PO1,PO2,PO3, PO8,PO10	Q1,Q3	15
CLO02	Apply object-oriented programming constructs such as classes, objects, constructors, functions, pointers, recursion and inheritance to solve problems	PO1,PO2, PO4,PO5	Q1,Q2,Q3,Q6	30
CLO03	Demonstrate understanding of algorithm complexity and perform basic operations on data structures such as arrays and linked lists.	PO3,PO10,PO11	Q2,Q4	30
CLO04	Understand the basics of data structure, complexity of algorithms, and the implementation of various operations on arrays and linked lists	PO1,PO2,PO9,PO11	Q1,Q2,Q4,Q6	25
CLO05	Implement searching and sorting algorithms for efficient data processing.	PO3,PO4, PO6, PO7, PO8, PO11	Q1,Q2,Q3	22
CLO06	Illustrate the concept of stack, linked list and queue data structures	PO1,PO3,PO5,PO6,PO10,PO11	Q1,Q2, Q3	28
Total Contact Hours				150

---

[1] National Higher Education Qualification Framework Level, Refer to annexure

[2] NHEQF Level Descriptor, Refer to Annexure & Learning outcomes descriptors for qualification for all levels on the NHEQF

[3] Types of Assessments can be referred from Type of Assessments. Refer to Annexure.

[4] For SDG Mapping with Courses, Pl refer SDG Mapping policy for Courses

24CSE0208/Data Structures using Object Oriented Programming

CLO-PO Mapping

CLO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	Type of Assessment's[3]
CLO01	H	M	M					L		H		Formative, Summative
CLO02	H	H		L	M							Formative, Summative
CLO03			M							M	L	Formative, Summative
CLO04	M	M							M		M	Formative, Summative
CLO05			M	M		L	M	L			H	Formative, Summative
CLO06	M		M	L		L				M	M	Formative, Summative

H=High, M=Medium, L=Low

3. **Recommended Books:**

> **B01:** Object Oriented Programming with C++' by E Balagurusamy, 6th Edition, Tata McGraw Hill.
> **B02:** Data Structures and Algorithms in Java, Robert Lafore, Sams Publishing, 2nd edition, 2002
> **B03:** Data structures and algorithms in Java. John, Goodrich MT, Tamassia R, Goldwasser MH, wiley 2014
> **B04:** Data Structure, Algorithms and Applications Using C++, S. Sahni, Sillicon Press, 2nd edition, 2005, 2014

Reference Books:

> **RB01:** Introduction to Algorithms by Thomas H. Cormen, The MIT Pressman 3rd Edition, 2001
> **RB02:** The textbook Algorithms, 4th Edition by Robert Sedgewick and Kevin Wayne, Pearson Education, Inc., 2011

4. **Other readings and relevant websites:**

Serial No	Link of Journals, Magazines, websites and Research Papers
1.	https://nptel.ac.in/courses/106105225
2.	https://cse.iitkgp.ac.in/~dsamanta/javads/index.htm
3.	http://www.tenouk.com/cncplusplustutorials.html
4.	https://algs4.cs.princeton.edu/home/
5.	https://library.chitkara.edu.in/subscribed-books.php

5. **Recommended Tools and Platforms**
   - Visual Studio IDE
   - Tespad

6. **Course Plan: Theory+ Lab Plan**

   Theory Plan

Lect. No.	Topic(s)
01-02	Introduction to basic concepts of object-oriented programming, key features, uses, and importance. Comparison between the procedural programming paradigm and object-oriented programming paradigm
03-04	Basic Operation in C++:- Arithmetic Operators, Logical operators, Bitwise operators, Control Flow & Statements
05-07	Defining Structure and classes in C++:- Introduction to Name Space, Specifying a class, Creating class objects, Accessing class members
08-11	Inheritance in C++ :- introduction, defining derived classes, forms of inheritance(single, multilevel, multiple), Access specifiers – public, private, and protected (Explain with inheritance)
12-18	Functions in C++: Call by value, Call by reference, Call by pointer, Return by reference, Function overloading, Lambda functions, Static members, Static objects, Constant member function, Constant objects, Friend functions, Friend class
19-25	Recursion in C++, Need for Constructors and Destructors.
26-29	Understanding pointers, Accessing address of a variable, Declaring & initializing pointers, Accessing a variable through its pointer
30-35	Data Structures and Algorithms: Importance in programming and real-world applications, Elementary Data Organization, Data Structure Types and Operations Types: Linear vs Non-linear, Static vs Dynamic.
36-42	Algorithm: Complexity Analysis, Time vs Space trade-offs, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o) Analysis, Operation counting, Iterative approach, Master theorem, Practice Problems for complexity computation
ST-1	
43-44	History and Features of Java, Java Virtual Machine (JVM), JRE and JDK, Setting up JAVA environment for (MacOS/Window/Linux), working on Integrated Development Environment(IDE)- Eclipse/NetBeans/Visual Studio code
45-48	Compiling and Interpreting Java Program, Understanding (public static void main(string[] arg) method command Line Arguments, Java Basics: Identifiers, Keywords, Java Data Types & Operators, Practice Problem covering all Java Operators
49-54	Control Statements in Java: Decision Constructs, using loop constructs, jump statement e.g break, continue and return. Practice problems using loops, Nested Loops and conditional statements, involving patterns
55-59	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays. Processing Multi-Dimensional Arrays as Collection of 1-D arrays, Applications in databases, caching, and matrix operations. Recursion using Java and its applications
60-68	Searching: Linear and Binary Search with their Complexity Analysis
69-72	Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort

73-76	Strings, StringBuffer StringBuilder & StringTokenizer: Difference between Character Array and String in Java, Creating Strings, Immutable property of String, Methods of String class, String Buffer class & StringBuilder class, toString method, StringTokenizer class, Practice Problems involving all String Class
ST-2 (Lec no. 43-76)	
77-85	Classes & Objects: Defining Classes, Access Control, Constructing Objects through Constructors, Default constructors, Parameterized constructors, Copy constructors, Wrapper Classes
86-90	Class variables (static keyword), Instance variables and methods, this keyword, Method overloading(Constructor overloading), Practice Problems
91-98	Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting elements from Linked List, Operations on Doubly Linked List, Circular linked List & its applications
99-101	Stacks: Array representation of Stacks, implementation of stack, using linked list.
102-107	Applications of Stack: Application in undo operations, Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks
108-112	Queues: Representation as Array and Linked List in Queue, Practice Problems
ST3 (Lec no. 43-112)	
113-120	Deques, Circular Queues, Priority Queues, Operations: insertion, deletion, and updation in the Queue
End Term Examination (ETE-Complete Syllabus)[#]	

#Coding questions need to be attempted only using JAVA

7. **Delivery/Instructional Resources Theory Plan:**

Lec. No.	Topics	CLO	Book No, CH No, Page No	TLM	ALM	Web References	Audio-Video
01	Introduction to basic concepts of object-oriented programming, key features, uses, and importance. Comparison between the procedural programming paradigm and object-oriented programming paradigm	1	B1 Chapter 1	Lecture Discussion Demonstration method using a simulation or a tool	Assignment	http://www.cplusplus.com/doc/tutorial/	https://onlinecourses.nptel.ac.in/noc24_cs125/preview
02	Basic Operation in C++:- Arithmetic Operators, Logical operators, Bitwise operators, Control Flow & Statements	1, 2	B1 Chapter 2,3	Discussion Demonstration method using a simulation or a tool	Lab Challenge Assignment	https://code.visualstudio.com/docs/setup/windows	https://code.visualstudio.com/docs/cpp/config-mingw
03-04	Defining Structure and classes in C++:- Introduction to Name Space, Specifying a class, Creating class objects, Accessing class members	1, 2	B1 Chapter 4,5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cplusplus.com/doc/tutorial/	https://nptel.ac.in/courses/106105151
05-06	Inheritance in C++ :- introduction, defining derived classes, forms of inheritance(single, multilevel, multiple), Access specifiers – public, private, and protected (Explain with inheritance)	2	B1 Chapter 7	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cprogramming.com/tutorial/c++-tutorial.html	https://nptel.ac.in/courses/106105151
07	Functions in C++: Call by value, Call by reference, Call by pointer, Return by reference, Function overloading, Lambda functions, Static members, Static objects, Constant member function, Constant objects, Friend functions, Friend class	2	B1 Chapter 6,9	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cprogramming.com/tutorial/c++-tutorial.html	https://nptel.ac.in/courses/106105151
08	Recursion in C++, Need for Constructors and Destructors	2,3	B1 Chapter 6	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.tenouk.com/cncplusplustutorials.html	https://nptel.ac.in/courses/106105151
09	Understanding pointers, Accessing address of a variable, Declaring & initializing pointers, Accessing a variable through its pointer	2,3	B1 Chapter 5,10	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cprogramming.com/tutorial/c++-tutorial.html	https://nptel.ac.in/courses/106105151

10-11	Data Structures and Algorithms: Importance in programming and real-world applications, Elementary Data Organization, Data Structure Types and Operations Types: Linear vs Non-linear, Static vs Dynamic.	2,3	B02: Chapter 1, B04: Chapter 1	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cprogramming.com/tutorial/c++-tutorial.html	https://nptel.ac.in/courses/106105151
12	Algorithm: Complexity Analysis, Time vs Space trade-offs, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o) Analysis, Operation counting, Iterative approach, Master theorem, Practice Problems for complexity computation	2,3	B02: Chapter 1, B03: Chapter 3, B04: Chapter 1	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cprogramming.com/tutorial/c++-tutorial.html	https://nptel.ac.in/courses/106105151
ST-1							
13	History and Features of Java, Java Virtual Machine (JVM), JRE and JDK, Setting up JAVA environment for (MacOS/Window/Linux), working on Integrated Development Environment(IDE)-Eclipse/NetBeans/Visual Studio code	3	B02: Chapter 1, B03: Chapter 1	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.tenouk.com/cncplusplustutorials.html	https://nptel.ac.in/courses/106105151
14	Compiling and Interpreting Java Program, Understanding (public static void main(string[] arg) method command Line Arguments, Java Basics: Identifiers, Keywords, Java Data Types & Operators, Practice Problem covering all Java Operators	1	B02: Chapter 2, B03: Chapter 2	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.tenouk.com/cncplusplustutorials.html	https://nptel.ac.in/courses/106105151
15-16	Control Statements in Java: Decision Constructs, using loop constructs, jump statement e.g break, continue and return. Practice problems using loops, Nested Loops and conditional statements, involving patterns	3	B02: Chapter 3, B03: Chapter 3	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	http://www.cplusplus.com/doc/tutorial/	https://nptel.ac.in/courses/106105151
17-18	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays. Processing Multi-Dimensional Arrays as Collection of 1-D arrays, Applications in databases, caching, and matrix operations. Recursion using Java and its applications	4	B02: Chapter 4, B03: Chapter 3	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://algs4.cs.princeton.edu/home/	https://nptel.ac.in/courses/106102064
19-20	Searching: Linear and Binary Search with their Complexity Analysis, Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort.	4	B02: Chapter 5, B03: Chapter 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://algs4.cs.princeton.edu/home/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-2-data-structures-and-dynamic-arrays/
21	Strings, StringBuffer StringBuilder & StringTokenizer: Difference between Character Array and String in Java, Creating Strings, Immutable property of String, Methods of String class, String Buffer class & StringBuilder class, toString method, StringTokenizer class, Practice Problems involving all String Classes	4,5	B02: Chapter 7, B03: Chapter 1	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://ocw.mit.edu/courses/1-204-computer-algorithms-in-systems-engineering-spring-2010/8ee75d49f1cb9a947f1d3f15a2aa9e00_MIT1_204S10lec05.pdf	https://nptel.ac.in/courses/106102064
ST-2							

	Topic	CLO	Book/Chapter	TLM	ALM	Web References	Audio-Video
22-23	Classes & Objects: Defining Classes, Access Control, Constructing Objects through Constructors, Default constructors, Parameterized constructors, Copy constructors, Wrapper Classes	4,5	B02: Chapter 6, B03: Chapter 2	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://algs4.cs.princeton.edu/home/	https://nptel.ac.in/courses/106102064
24-25	Class variables (static keyword), Instance variables and methods, this keyword, Method overloading(Constructor overloading), Practice Problems	4,5	B02: Chapter 6, B03: Chapter 2	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://algs4.cs.princeton.edu/home/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-5-linear-sorting/
26	Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting elements from Linked List, Operations on Doubly Linked List, Circular linked List & its applications	6	B02: Chapter 6, B03: Chapter 5, B04: Chapter 3	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://cse.mait.ac.in/pdf/LAB%20MANUAL/DS.pdf	https://nptel.ac.in/courses/106102064
27	Stacks: Array representation of Stacks, implementation of stack, using linked list. Applications of Stack: Application in undo operations, Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks	7	B02: Chapter 7, B03: Chapter 6, B04: Chapter 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://cse.mait.ac.in/pdf/LAB%20MANUAL/DS.pdf	https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/resources/session-15-static-trees/
28	Applications of Stack: Application in undo operations, Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks	7	B02: Chapter 7, B03: Chapter 6, B04: Chapter 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://cse.mait.ac.in/pdf/LAB%20MANUAL/DS.pdf	https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/resources/session-15-static-trees/
29	Queues: Representation as Array and Linked List in Queue, Practice Problems	7	B02: Chapter 8, B03: Chapter 6, B04: Chapter 5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://cse.mait.ac.in/pdf/LAB%20MANUAL/DS.pdf	https://nptel.ac.in/courses/106102064
ST3							
30	Deques, Circular Queues, Priority Queues, Operations: insertion, deletion, and updation in the Queue	7	B02: Chapter 8, B03: Chapter 6, B04: Chapter 5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://cse.mait.ac.in/pdf/LAB%20MANUAL/DS.pdf	https://nptel.ac.in/courses/106102064
End Term Examination (ETE)							

Lab Plan:

Lab No.	Experiment	CLO	TLM	ALM	Web References	Audio-Video
01-02	Factorial of a number Swap two numbers without using third one Sum of a set of numbers	1	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/program-for-factorial-of-a-number/ https://www.geeksforgeeks.org/swap-two-numbers-without-third-variable-in-cpp/ https://www.geeksforgeeks.org/cpp-program-to-find-sum-of-first-n-natural-numbers/
03-04	Functions to Display String Print Kth largest number	1, 2	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.programiz.com/cpp-programming/strings https://www.geeksforgeeks.org/k-largestor-smallest-elements-in-an-array/

05-07	Write a C++ program to declare a class . Declare pointer to class. Initialize and display the contents of the class members	1, 2	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/cpp/c-classes-and-objects/
08-11	Implementing inheritance in C++ Modes of Inheritance	2	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/cpp/inheritance-in-c/
12-18	Function Overloading, Call by Value, Call by Reference Recursive Function and Lambda Function	2,3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/function-overloading-c/ https://www.geeksforgeeks.org/lambda-expression-in-c/
19-23	Write a program to show the working of static member functions	2,3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/cpp/static-member-function-in-cpp/
24-30	Concept of a Pointer, Declaring and Initializing Pointer Variables and Pointer Expressions	2,3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/c/c-pointers/
31-37	Constructors, Destructors & Friend Function	3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/cpp/friend-class-function-cpp/
38-41	Pointers in C++, Accessing the Address of a Variable:	2,3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/cpp/cpp-pointers/
42-43	Operations of Data Structures	4	Demonstration method using a simulation or a tool	Lab Challenge	NA	
44-46	Algorithm Complexity and Complexity Computation Calculate Time and Space complexity of algorithms	4,5	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/time-complexity-and-space-complexity/
ST1						
47-51	Control Statements: Decision Constructs, Using Loop Constructs, Jump Statements e.g. break, continue and return	4	Demonstration control statements	Lab Challenge	NA	https://www.geeksforgeeks.org/loops-in-java/?ref=lbp
52-54	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays.	4	Demonstration array traversal and applications in matrix operations	Lab Challenge	NA	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-2-data-structures-and-dynamic-arrays/
55-56	Find out the largest and second largest element in array. Reverse the element of an array.	4	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/program-to-find-largest-element-in-an-array/
57-60	Perform Linear Search and Binary Search on an array. Search the element by passing the array to a function and then returning the position of the element from the function	4,5	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/linear-search/
61-67	Sorting techniques: Selection Sort, Insertion	4,5	Demonstration method using a simulation or a	Lab Challenge	NA	https://www.geeksforgeeks.org/sorting-algorithms/

	Sort, Quick Sort, Merge Sort. Implementation of selection sort, Implementation of Insertion sort, Implementation of Quick sort, Implementation of Merge sort		tool		
68-76	Strings, StringBuffer, StringBuilder & StringTokenizer: Difference between Character Array and String in Java, Creating Strings, Immutable property of String, Methods of String class, String Buffer class & StringBuilder class, toString method, String Tokenizer class Practice Problems involving all String Classes	5	Demonstration of All String Classes, Count vowels and consonants, Convert string to character array and back	Lab Challenge	NA	https://www.geeksforgeeks.org/strings-in-java/?ref=lbp
ST-2					
77-79	Classes & Objects: Defining Classes, Access Control, Constructing Objects through Constructors, Default constructors, Parameterized constructors, Copy constructors, Wrapper Classes	5	Creating object using default constructor, copy constructor Wrapper class demonstration	Lab Challenge	NA	https://www.geeksforgeeks.org/methods-in-java/?ref=dhm
80-82	Class variables (static keyword), Instance variables and methods, this keyword, Method overloading(Constructor overloading), Practice Problems	5	Constructor Overloading: with number of parameters	Lab Challenge	NA	https://www.geeksforgeeks.org/constructor-overloading-java/
83-86	Create a linked list with nodes having information about a student and perform Insert a new node at specified position. Delete of a node with the roll number of student specified. Reversal of that linked list	6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/student-record-management-system-using-linked-list/
87-92	Create a stack and perform Pop, Push, Traverse operations on the stack using Linear Linked list	6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/implement-a-stack-using-singly-linked-list/
93-98	Applications of Stack: Application in undo operations, Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks	6	Infix to Postfix Conversion Evaluate Postfix Expression Factorial using Non-Recursive Stack Approach	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/prefix-postfix-conversion/
99-108	Queues: Representation as	6	Queue	Lab	NA	https://www.geeksforgeeks.org/java/queue-implementation-

	Array and Linked List in Queue, Practice Problems		Implementation using Array, Queue Implementation using Linked List	Challenge		using-linked-list-in-java/?utm_source=chatgpt.com
108-112	The enqueue operation can be used to add the element to the rear of the queue.	6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/introduction-and-array-implementation-of-queue/
ST3						
113-120	The dequeue operation can be used to removes the element from the front of the queues	6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/introduction-and-array-implementation-of-queue/
End Term Examination (ETE-Complete Syllabus)						

8. Remedial Classes[3]

After every Sessional Test, identify weak learners, and prepare the student lists. The following methods would be used to improve the performance of students.

- Remedial Classes on Saturdays
- Offer supplementary materials or activities to reinforce concepts outside regular class hours.
- Special discussions would be planned and scheduled accordingly.

9. Self-Learning[4]

Assignments to promote self-learning, survey of contents from multiple sources.

S.No	Topics	CLO	ALM	References/MOOCS
1	Creating a new file and read the content of file	3	Think – Pair- Share	https://www.geeksforgeeks.org/c/basics-file-handling-c/
2	Operations on Circular linked List and its applications	6	Think – Pair- Share	https://archive.nptel.ac.in/courses/106/102/106102064/
3	Dynamic programming to find the nth Fibonacci Number	5,6	Think – Pair- Share	https://www.w3schools.com/dsa/dsa_ref_dynamic_programming.php

10. Delivery Details of Content Beyond Syllabus[5]

Content beyond syllabus covered (if any) should be delivered to all students that would be planned, and schedule notified accordingly.

S.No	Advanced Topics, Additional Reading, Research papers and any	CLO	POs	ALM	References/MOOCS
1	Dijkstra algorithm	7	1, 2, 3, 4, 5, 10	Brainstorming session	https://www.w3schools.com/dsa/dsa_algo_graphs_dijkstra.php https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/

[3] Refer to Annexure
[4] Refer to Annexure
[5] Refer to Annexure

11. Evaluation Scheme & Components:

Students need to score at least 50% marks in the Internal Evaluation (components 1 to 4) and 50% marks in the End Term Examination (component 5) separately to pass the course.

Assessment Type[6]	Evaluation Component[7]	Type of Component[8]	No. of Assessments[9]	% Weightage of Component	Max. Marks	Mode of Assessment	CLO
Formative	Component 1	Testpad module progress and completion [#]	--	05%	05	Online	1 – 6
Formative	Component 2	Formative Assessments (FA)	05[§]	10%	10	Online	1 – 6
Formative	Component 3	Sessional Tests (ST-1, ST-2)	02*	20%	20	Online	1 - 6
Formative	Component 4	ST-3	01**	15%	15	Offline	1- 6
Summative	Component 5	End Term Examination	01***	50%	50	Online	1- 6
	Total			**100%**			

[#] Any progress or attempt made on TestPad after the deadline will not be considered.

[§] There will be 05 formative assessments, and the scores of the BEST 03 will be considered.

* Students will have to appear in all the Sessional Tests.
*Makeup Examination will compensate for either ST-1 or ST-2 (Only for genuine cases, based on the Dean's approval).
**ST-3 No makeup exam will be taken for ST-3
***As per Academic Guidelines minimum 75% attendance is required to become eligible for appearing in the End Semester Examination.

---

[6] Refer to Annexure 2 of NCrF
[7] Refer to Annexure
[8] Refer to Annexure
[9] Refer to Annexure

24CSE0208/Data Structures using Object Oriented Programming

12. Syllabus of the Course:

S.No.	Topic(s)	No. of Lectures	Weightage %
	Subject: Data Structures using Object Oriented Programming		
1.	Introduction to basic concepts of object-oriented programming, key features, uses, and importance. Comparison between the procedural programming paradigm and object-oriented programming paradigm Setting up your Environment, Installing C++, Code Editors / IDEs Basic Operation- Arithmetic Operators, Logical operators, Bitwise operators, Control Flow & Statements. Defining Structure and classes. Introduction to Name Space, Specifying a class, Creating class objects, Accessing class members	11	8%
	Inheritance: introduction, defining derived classes, forms of inheritance(single, multilevel, multiple), Access specifiers – public, private, and protected (Explain with inheritance)	19	13%
	Functions in C++: Call by value, Call by reference, Call by pointer, Return by reference, Function overloading, Lambda functions, Static members, Static objects, Constant member function, Constant objects, Friend functions, Friend class, Recursion in C++, Need for Constructors and Destructors.	12	7%
	Understanding pointers, Accessing address of a variable, Declaring & initializing pointers, Accessing a variable through its pointer, Data Structures and Algorithms: Importance in programming and real-world applications, Elementary Data Organization, Data Structure Types and Operations Types: Linear vs Non-linear, Static vs Dynamic. Algorithm: Complexity Analysis, Time vs Space trade-offs, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o) Analysis, Operation counting, Iterative approach, Master theorem, Practice Problems for complexity computation	17	12%
2	History and Features of Java, Java Virtual Machine (JVM), JRE and JDK, Setting up JAVA environment for (MacOS/Window/Linux), working on Integrated Development Environment(IDE)- Eclipse/NetBeans/Visual Studio code, Compiling and Interpreting Java Program, Understanding (public static void main(string[] arg) method command Line Arguments, Java Basics: Identifiers, Keywords, Java Data Types & Operators, Practice Problem covering all Java Operators	12	5%
	Control Statements in Java: Decision Constructs, using loop constructs, jump statement e.g break, continue and return. Practice problems using loops, Nested Loops and conditional statements, involving patterns Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays. Processing Multi-Dimensional Arrays as Collection of 1-D arrays, Applications in databases, caching, and matrix operations. Recursion using Java and its applications	16	15%
	Searching: Linear and Binary Search with their Complexity Analysis Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort	17	10%
	Strings, StringBuffer StringBuilder & StringTokenizer: Difference between Character Array and String in Java, Creating Strings, Immutable property of String, Methods of String class, String Buffer class & StringBuilder class, toString method, StringTokenizer class, Practice Problems involving all String Class	15	10%
3	Classes & Objects: Defining Classes, Access Control, Constructing Objects through Constructors, Default constructors, Parameterized constructors, Copy constructors, Wrapper Classes Class variables (static keyword), Instance variables and methods, this keyword, Method overloading(Constructor overloading), Practice Problems, Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting elements from Linked List, Operations on Doubly Linked List, Circular linked List & its applications	11	7%
	Stacks: Array representation of Stacks, implementation of stack, using linked list. Applications of Stack: Application in undo operations, Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks	9	6%
	Queues: Representation as Array and Linked List in Queue, Practice Problems, Deques, Circular Queues, Priority Queues, Operations: insertion, deletion, and updation in the Queue	11	7%
	End Term Exam		

13. **Academic Integrity Policy:**

Education at Chitkara University builds on the principle that excellence requires freedom where Honesty and integrity are its prerequisites. Academic honesty in the advancement of knowledge requires that all students and Faculty respect the integrity of one another's work and recognize the importance of acknowledging and safeguarding intellectual property. Any breach of the same will be tantamount to severe academic penalties.

This Document is approved by:

Designation	Name	Signature
Course Coordinator	Dr. Heena Wadhwa	
Head-Academic Delivery	Dr. Mrinal Paliwal	
Dean	Dr. Rishu Chhabra	
Date(DD/MM/YYYY)	06/08/2025	

Annexure

1. **Pre- requisite**
   Mention The Pre-requisite skill set or course/s if it is expected to be studies before this course, otherwise write "not applicable".

2. **NHEQF levels**

The NHEQF levels represent a series of sequential stages expressed in terms of a range of learning outcomes against which typical qualifications are positioned/located. NHEQF level 4.5 represents learning outcomes appropriate to the first year (first two semesters) of the undergraduate programme of study, while Level 8 represents learning outcomes appropriate to the doctoral-level programme of study. Table 1: Higher education qualifications at different levels on the NHEQF

NHEQF level	Examples of higher education qualifications located within each level
Level 4.5	Undergraduate Certificate. Programme duration: First year (first two semesters) of the undergraduate programme, followed by an exit 4-credit skills-enhancement course(s).
Level 5	Undergraduate Diploma. Programme duration: First two years (first four semesters) of the undergraduate programme, followed by an exit 4-credit skills-enhancement course(s) lasting two months.
Level 5.5	Bachelor's Degree. Programme duration: First three years (Six semesters) of the four-year undergraduate programme.
Level 6	Bachelor's Degree (Honours/ Honours with Research). Programme duration: Four years (eight semesters).
Level 6	Post-Graduate Diploma. Programme duration: One year (two semesters) for those who exit after successful completion of the first year (two semesters) of the 2-year master's programme.
Level 6.5	Master's degree. (e.g. M.A., M.Com., M.Sc., etc.) Programme duration: Two years (four semesters) after obtaining a 3- year Bachelor's degree (e.g. B.A., B.Sc., B.Com.etc.).
Level 6.5	Master's degree. (e.g. M.A., M.Com., M.Sc., etc.) Programme duration: One year (two semesters) after obtaining a 4 -year Bachelor's degree (Honours/ Honours with Research) (e.g. B.A., B.Sc., B.Com. etc.).
Level 7	Master's degree.(e.g. M.E./M.Tech. etc.) Programme duration: Two years (four semesters) after obtaining a 4- year Bachelor's degree. (e.g. B.E./B.Tech. etc.)
Level 8	Doctoral Degree

3. **NHEQF level descriptors**

Each NHEQF level is structured based on the defined learning outcomes which lead to the expected graduate attributes/profile. The level descriptors reflect the expected outcomes of learning that should be achieved and demonstrated by graduates of a specific programme of study leading to a qualification at a specific NHEQF level.

Click Learning outcomes descriptors for qualification for all levels on the NHEQF

4. **Course Outcomes**
   The number of Course Outcomes is recommended to be 4-5 for courses that do not contain practical component and 6 for those courses with a practical component. Flexibility can be sought by the post-graduate courses in this regard.

5. **Theory/lab Plan**
   The following are the guidelines to be followed while creating plans
   - Each session may be planned for a duration of 45/50mins (irrespective of the double hour or single hour scheduled in timetable).
   - Every session must incorporate at least one active learning method which may or may not be part of the assessments.
   - Put BoS Approved Syllabus in the topics. Deviations (if any) from BoS approved syllabus must be brought to the notice of BoS chairman & Dean Academics, After approval, revised handout should be submitted.
   - The Topics elaborated in the Theory/Lab plan must match those in the course execution plan.

6. **Teaching Learning Methods**
   The following are some of the Teaching & Learning methods that can be incorporated in session wise teaching learning plan.
   - **Teacher-centered Learning Methods:**
     i. Lecture
     ii. Discussion
     iii. Demonstration method using a simulation or a tool
     iv. Reviewing
     v. Questioning

   - **Learner-centered teaching & Learning methods:**
     i. **Active learning**, in which students solve problems, answer questions, formulate questions of their own, discuss, explain, debate, or brainstorm during class;
     ii. **Cooperative learning**, in which students work in teams on problems and projects under conditions that assure both positive interdependence and individual accountability; and
     iii. **Inductive teaching and learning**, in which students are first presented with challenges (questions or problems) and learn the course material in the context of addressing the challenges.

iv. **Inductive methods** include inquiry-based learning, case based instruction, problem-based learning, project-based learning, discovery learning, and just-in-time teaching. It is important to integrate authentic, reflective and collaborative learning experiences when designing for student-centered learning.

7. **Active Learning Methods**

The following are some of the Active Learning Methods that can be incorporated in session wise teaching learning plan.

- One Minute Paper
- Group Discussion
- Student-Created PPT, Charts, Matrices, Flowcharts, Models
- The Fish Bowl
- Debate
- Video Synthesis
- Quiz/Test Questions
- Brain Storming Sessions
- Case Study
- Shadowing
- Leading Question
- Puzzle, Enigma, Contradiction
- Statement-Opinion-Summary
- Think / Pair / Share
- Peer Review
- Just in Time Teaching
- Statement-Opinion-Summary
- Peer Survey
- Focused Listing
- Role-Playing
- Student Field Work with Reflection
- Infusing Humor into Class Sessions
- Inviting Effective Guest Speakers

8. **Remedial Classes**

After every Sessional Test, identify weak learners, provide supplement course handout. Student list and Impact Observed report should be submitted to Dean through proper channel.

9. **Self Learning**

Plan 10% of topics in self-learning mode with discussions, ALM's and Assessment happing in the class.

10. **Content Beyond Syllabus**

Plan Advanced Topics, Experiments, Additional Reading, Research papers in self-learning mode with ALM's and Assessment happing in the regular class or lab. Usually caters advanced learners. Identify Advanced learners. For Extra classes, schedule should be notified accordingly.

11. **Assessment Type**

1. Assessment broadly can be classified into the following types:

a. **Diagnostic assessments**: Diagnostic assessments are intended to help teachers identify what students know and can do in different domains to support their students' learning. These help teachers determine strengths of students in various areas to better address their specific needs.

b. **Formative assessments**: Formative assessment refers to a wide variety of methods that teachers use to conduct in-process evaluations of student comprehension, learning needs, and academic progress during a lesson, unit, or a course. Formative assessments help teachers identify concepts that students are struggling to understand, skills they are having difficulty acquiring, or learning standards they have not yet achieved so that adjustments can be made to lessons, instructional techniques, and academic support.

c. **Summative assessments**: Summative assessment is an assessment administered at the end of an instructional unit in a course. These assessments are intended to evaluate student learning by comparing performance to a standard or benchmark.

d. **Ipsative assessments**: Ipsative assessment involves comparisons between past and current work to identify a learner's growth over time, rather than progress toward an external set of criteria. Therefore, Ipsative assessment is an internal or self-referenced assessment.

e. **Norm-referenced assessments**: Norm-referenced tests report whether test takers performed better or worse than a hypothetical average student, which is determined by comparing scores against the performance results of a statistically selected group of test takers, typically of the same age or grade level, who have already taken the exam.

f. **Criterion-referenced assessments**: Criterion-Reference tests measure the performance of test takers against the criteria covered in the curriculum.

g. **Peer-to-Peer randomised Assessments**: Peers will be able to provide assessment in this case

h. **Industry Validation of Effectiveness**: In the Vocation Education, Industry validation of effectiveness of training is particularly important.

i. **Self-assessments**: To evaluate how much the learner has grasped by self-learning.

2. Other Assessment Methods: Conducting an assessment takes time, thought, attention, planning, and often collaboration. Each assessment tool, whether a short survey or detailed rubric, will be useful only insofar as it both addresses the outcomes well and is feasible to use.

a. **Rubrics**: For assessing qualitative student work such as essays, projects, reports, or presentations. Rubrics serve well to clearly denote the specific expectations for an assignment, for collecting data for assessment of student learning outcomes. and for student performance. Rubrics can be used for grading, for providing feedback to students, and for informing and encouraging students to think about their own learning.

b. **Portfolios and E-Portfolio**: Portfolios can provide a window into the process of student learning across a semester-long project that can be assessed (usually by using a rubric).

c. **Curriculum Mapping**: A good curriculum map can serve to focus assessment, and the improvements that follow, where it will be most useful, informative, or effective.

d. **Structured Interviews**: While time-consuming, structured interviews are useful when specific questions need to be asked. It also leaves room for unplanned topics or ideas to emerge.

e. **Student Experience Surveys**: Student experience in research universities (SERU), including administration of on-line census SERU Undergraduate and Graduate Surveys, can yield important information about student perceptions and experiences.

12. **Evaluation Component & Types**

As per LMs we need to figure it out whether it is component 1, 2 or 3. In Types of Evaluation Component, we need to specify what type of evaluation we are performing like Continuous Evaluation or Sessional Test or End Term Examination.

13. **No. of Assessments and Weightage of Components**

Department will give guideline for number of assessments, mandatory or optional and weightage.