UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

COURSE PLAN

Target	50% (marks)
Level-1	40% (population)
Level-2	50% (population)
Level-3	60% (population)

1. Method of Evaluation

UG
Viva voce and Quiz (50%)
Performance & Records (50%)

2. Passing Criteria

Scale	UG
Out of 10 point scale	SGPA – "5.0" in each semester CGPA – "5.0" Min. Individual Course Grade – "C" Course Grade Point – "4.0"

*for UG, passing marks are 35/100 in a paper

3. Pedagogy

Solution to the problems should be designed (algorithm/flow-chart/pseudocode) and tested. After obtaining a successful design, the solution to the problem is implemented using C++ language. Students are evaluated based on Performance (via efficient design, implementation) and record keeping, and preparation of students (via viva-voce and quiz).

4. References:

Text Books	Web resources	Reference books
1. Herbert Schildt, "C++: The Complete Reference", McGraw Hill Education, 2003. 2. John R. Hubbard, "Data Structures with C++", Schaum's Outlines, Tata McGraw Hill Education, 2000.	NPTEL: https://youtu.be/LZFoktwiars	1. Michael T. Goodrich, Roberto Tamassia, David Mount, "Data Structures and Algorithms in C++", Wiley India Pvt. Ltd., 2004. 2. Seymour Lipschutz, "Data Structures", Schaum's Outlines, Tata McGraw Hill Education, 2006.

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

GUIDELINES TO STUDY THE SUBJECT

Instructions to Students:

1. Go through the 'Syllabus' in the Black Board section of the web-site(https://learn.upes.ac.in) in order to find out the Reading List.
2. Get your schedule and try to pace your studies as close to the timeline as possible.
3. Get your on-line lecture notes (Content, videos) at Lecture Notes section. These are our lecture notes. Make sure you use them during this course.
4. Check your blackboard regularly
5. Go through study material
6. Check mails and announcements on blackboard
7. Keep updated with the posts, assignments and examinations which shall be conducted on the blackboard
8. Be regular, so that you do not suffer in any way
9. Cell Phones and other Electronic Communication Devices: Cell phones and other electronic communication devices (such as Blackberries/Laptops) are not permitted in classes during Tests or the Mid/Final Examination. Such devices MUST be turned off in the class room.
10. E-Mail and online learning tool: Each student in the class should have an e-mail id and a pass word to access the LMS system regularly. Regularly, important information – Date of conducting class tests, guest lectures, via online learning tool. The best way to arrange meetings with us or ask specific questions is by email and prior appointment. All the assignments preferably should be uploaded on online learning tool. Various research papers/reference material will be mailed/uploaded on online learning platform time to time.
11. Attendance: Students are required to have minimum attendance of 75% in each subject. Students with less than said percentage shall NOT be allowed to appear in the end semester examination.

This much should be enough to get you organized and on your way to having a great semester! If you need us for anything, send your feedback through e-mail to your concerned faculty. Please use an appropriate subject line to indicate your message details.

There will no doubt be many more activities in the coming weeks. So, to keep up to date with all the latest developments, please keep visiting this website regularly.

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

RELATED OUTCOMES

1. The expected outcomes of the Program are:

PO1	Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	Individual and team-work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	Communication: Communicate effectively on complex engineering activities with the engineering community and with society at-large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

2. The expected outcomes of the Specific Program are:

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma — Course Code: CSEG1104
2. Course : Advanced Data Structures Lab — L: 0
3. Program : B.Tech. CSE spl. in AI & ML — T: 0
4. Target : 50% — P: 1, C: 1

PSO1	Perform system and application programming using computer system concepts, concepts of Data Structures, algorithm development, problem solving and optimizing techniques.
PSO2	Apply software development and project management methodologies using concepts of front-end and back-end development and emerging technologies and platforms.
PSO3	To create & develop most efficient solutions by applying machine learning with analytical emphasis on industrial and research problems.

3. The expected outcomes of the Course are:

CO1	Demonstrate a clear understanding of the C++ programming environment and basic concepts including encapsulation, access control and data hiding, constructors, destructors and dynamic memory allocation.
CO2	Design solutions using the concepts of Inheritance and Polymorphism in object-oriented programming.
CO3	Exhibit a clear understanding of Generic Programming, Exception Handling and File Handling.
CO4	Demonstrate and develop skills to work with Hash Tables, Trees and Graph data structures.

4. CO-PO/PSO Relationship Matrix

PO/PSO / CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	2	1	0	0	0	0	0	0	0	1	3	0	0
CO2	1	1	2	1	0	0	0	0	0	0	0	1	3	0	0
CO3	1	1	2	1	0	0	0	0	0	0	0	1	3	0	0
CO4	1	1	2	1	0	0	0	0	0	0	0	1	3	0	0
Average	1	1	2	1	0	0	0	0	0	0	0	1	3	0	0

1- Slight (low) 2- Moderate (Medium) 3-Substantial (high)

5. Course Outcomes assessment plan:

Components / Course Outcomes	Quiz-1	Quiz-2	Viva-1	Viva-2
CO1	✔		✔	
CO2	✔		✔	
CO3		✔		✔
CO4		✔		✔

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

BROAD PLAN OF COURSE COVERAGE

Course Activities:

S.No.	Description	Planned		Remarks
		Batch-5	No. of Sessions	
1.	Classes and Objects	14-Jan-20	1	CO1
2.	Constructors and Destructors	21-Jan-20	1	CO1
3.	Dynamic Memory Allocation	28-Jan-20	1	CO1
4.	Friends to a Class and Inheritance	4-Feb-20	1	CO2
5.	Redefinition and Overloading	11-Feb-20	1	CO2
6.	Virtual Functions and Abstract Classes	18-Feb-20	1	CO2
7.	Generic Programming	10-Mar-20	1	CO3
8.	File Handling	17-Mar-20	1	CO3
9.	Exception Handling	24-Mar-20	1	CO3
10.	Hash Tables	31-Mar-20	1	CO4
11.	Trees	7-Apr-20	1	CO4
12.	Graphs	14-Apr-20	1	CO4

Sessions: Total No. of Instructional periods available for the course

SESSION PLAN

Experiment-1: Classes and Objects

Objective: To implement the concept of classes and objects, encapsulation, abstraction and data hiding while applying the different operators, sequential and looping constructs.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Consider rectangle, square, triangle and circle as 4 given shapes with a common area of 314 sq.cm. The length of a rectangle is 2cm and height of right-angled triangle is 4cm. Find the shape which has the smallest perimeter and largest perimeter.
2. Use for loop to print the truth table for the expression XY+Z.
3. Convert Decimal number into a Binary Number using 1D array.

X	Y	Z	XY+Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
.	.	.	.
.	.	.	.
1	1	1	1

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

Experiment-2: Constructors and Destructors

Objective: To show the use of Constructors in object instantiation and use of Destructor in freeing the resources that the object may have acquired during its lifetime while realizing the concepts of reference variables, recursive functions and arrays used to implement Queue data structure.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Design a class with a recursive member function with prototype “int fact(int &no)” to find the factorial of a number (n) obtained using a parameterized constructor. The class also has a destructor.
2. Implement the operations of a Queue Data Structure using 1D array and constructor initialization list.

Experiment-3: Dynamic Memory Allocation

Objective: To illustrate dynamic allocation of memory in the implementation of Stack data structure with linked list and dynamic memory allocation for objects, and application of this pointers in C++.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Implement the operations of a dynamically allocated Stack data structure using a self-referential structure and single Linked List.
2. Create a C++ class with member functions that use this pointer to perform complex number arithmetic. The objects of this class should be dynamically allocated [Hint: use private destructor].

Experiment-4: Friends to a Class and Inheritance

Objective: To demonstrate the use of Friend functions, Friend Class, and application of Inheritance.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

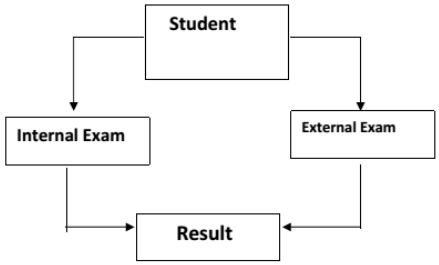
UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

1. Class "Student" has members functions: student_Details() and display_Student_Details() and data members: name, roll no, and gender. The classes "Internal_Exam" and "External_Exam" have separate functions to get and display Internal and External marks for 'n' subjects. Define Class "Result" with functions to evaluate and display the results calculated with 40% internal marks + 60% external marks. Implement Inheritance as shown in the diagram.



2. Obtain molecular values for Oxygen, Hydrogen and Sulphur in three different classes named "Oxygen", "Hydrogen" and "Sulphur". These classes have a common friend function used to find out the chemical compounds (like Sulphuric acid, Sulphurous acid, Thiosulphuric acid) that can be formed based on the given input molecular values.
3. Class "Student" has data members: sapid, name, and marks and suitable member functions to obtain the students details and marks of six subjects. Use friend class to access the details of a student and calculate the total and average marks and prints the result.

Experiment-5: Redefinition and Overloading

Objective: To apply the concepts of function redefinition, function overloading and operator overloading.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Define a class with its constructor to obtain side1 and side2 and member functions to do the following:
   - void choice() - menu driven control to find and print the area of square or rectangle or triangle
   - float calculate(int) - compute the Area of Square
   - float calculate(int, int) - compute the Area of Rectangle
   - float calculate(float, float) - compute the Area of Triangle
2. Add two matrices by overloading binary operator +.
3. Base class "Temperature" obtains temperature value in Celsius (c) through its parameterized constructor with default arguments and convert it into Fahrenheit value (f) and prints it using the function "calculate()". Class "Temp" inherits class "Temperature" to obtain the temperature value in Fahrenheit and converts it into Kelvin value (k) and prints it using the redefined function "calculate()".

Experiment-6: Virtual Functions and Abstract Classes

Objective: To illustrate the use of virtual functions and pure virtual functions to realize function overriding and abstract classes.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

1. Base class "Temperature" obtains temperature value in Kelvin (k) through its parameterized constructor, convert it into Celsius value (c) and prints it using the virtual function calculate(). Class "Temp" inherits class "Temperature" to obtain the temperature value in Celsius (c), converts it into Fahrenheit value (f) and prints it using the overridden function calculate().
2. Abstract Base class "Temperature" obtains temperature value in Celsius (c) through its constructor and has a pure virtual function calculate(). Class "Fahrenheit" inherits class "Temperature" to obtain the temperature value in Celsius (c), converts it into Fahrenheit value (f) and prints it using calculate(). Class "Kelvin" inherits class "Temperature" to obtain the temperature value in Celsius (c), converts it into Kelvin value (k) and prints it using calculate().

Experiment-7: Generic Programming

Objective: To demonstrate the application of Parametric Polymorphism while working with function templates and class templates.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Write a generic function to sort the given elements in ascending order using bubble sort algorithm. This generic function must support integer and float types.
2. Write a generic class to implement the operations of a stack data structure using arrays. This generic class must support integer and float types.

Experiment-8: File Handling

Objective: To demonstrate file normal access and random access while working with simple file handling applications.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Write contents into a Test file. Get a word from the user as input, compute the number of occurrences of this word in the file and print the count.
2. Define class "Residence" with data members: house_no, income, house_name, and house_type. The class has the following member functions:
    - void input() – obtains house_no, house_name and income. Calls assign()
    - void assign() – allocates house type based on income (A/B/C/etc.)
    - void output() – prints house details
    - int house() – returns house_no

1. Name of Faculty : Vivek Shahare, Richa Choudhary, Amit Verma
2. Course : Advanced Data Structures Lab
3. Program : B.Tech. CSE spl. in AI & ML
4. Target : 50%

Course Code: CSEG1104
L: 0
T: 0
P: 1
C: 1

main() is menu driven to (1) add multiple records as required into a binary file in append mode (2) Enter house_no to locate the record in the binary file and modify it using random access (3) display records from the binary file.

Experiment-9: Exception Handling

Objective: To show the use of try-catch block, throw and rethrow statements in exception handling while working with simple C++ applications and also to demonstrate working with user-defined exceptions.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Write an interactive program to compute different roots of a Quadratic Equation. While computing square root of a number the input value must be tested for validity. If it is negative, the user defined function my_sqrt() should be invoked by set_terminate() to handle the exception promptly.
2. Write an interactive program to compute different roots of a Quadratic Equation. While computing square root of a number the input value must be tested for validity. If it is negative, the catch block of the user-defined exception class object should be made to handle the exception promptly apart from the other catch blocks.
3. Illustrate Stack unwinding while working with Exception Handling.

Experiment-10: Hash Tables

Objective: To implement hash tables with and without collision avoidance algorithms using arrays/linked lists.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Implement a hash function on student SAP-ID and categorize them in to their 10 families based on the last three digits. Example: Student with SAP-ID 5000423 belongs to family 9 and student with SAP-ID 5000425 belongs to family 2 based on last three digits.
2. Implement a Hash table using arrays. Perform Insert, Delete and Search operations on the hash table using the above Hash function (S.No.1). Adopt a suitable user-defined exception handling strategy if collision occurs while inserting data.
3. Implement a Hash table using arrays. Perform Insert, Delete and Search operations on the hash table using the above Hash function (S.No.1) and with Linear probing as Collision avoidance strategy.

Experiment-11: Trees

Objective: To demonstrate creation of a binary tree using arrays/linked lists and working with tree traversal and heap sorting algorithms.

UPES
UNIVERSITY WITH A PURPOSE

Year : 2019-20
Semester : II

1.	Name of Faculty	: Vivek Shahare, Richa Choudhary, Amit Verma	Course Code: CSEG1104
2.	Course	: Advanced Data Structures Lab	L: 0
3.	Program	: B.Tech. CSE spl. in AI & ML	T: 0
4.	Target	: 50%	P: 1
			C: 1

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Create a binary tree using an array/linked List.
2. Construct a Binary Tree and perform Inorder, Preorder and Postorder Traversal.
3. Implement Heap Sort.

Experiment-12: Graphs

Objective: To show the representation of graphs using adjacency matrix.

List of Lab Activities:

Write algorithm, prepare test cases and dry run the algorithm to test the design. Code using C++ language and evaluate the code using the test cases so as to obtain the desired results.

1. Accept the vertices and edges for a graph and stores it as an adjacency matrix. Implement functions to print in-degree and out-degree of any vertex 'i'. Also display the adjacency matrix.
2. Accept the graph as an adjacency matrix and check if the graph is undirected. [Hint: The matrix for undirected graph is symmetric.]