****

**Format No. QSP/7.1/01.F01 (B) Issue No.05 Rev. No 5 Dated: Jan 1, 2017**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**College of Engineering Studies**

**Dehradun**

**COURSE PLAN**

Programme : B. Tech CSE

Course : Advanced Data Structures

Subject Code : SEG-1004

No. of credits : 4

Semester : II

Session : 2017-18

Batch : 2016-20

Prepared by : Prakash GL

Email : prakashgl@ddn.upes.ac.in

Co-Faculties: Dr. Venkatadari, S. Christalin Nelson, Mr Hukam Singh

Rana, Mr Amit Singh,Sandip kumar Chaurasiya,

Mr. Amar Shukla,Mr. Alind,Amber Hayat,Bhupesh Devangan[B3+B4],Rashmi Sharma[B5+B6], Dr shamik Tiwari, Ms. Kalpana

**Approved By**

Faculty HOD/ Prog. Head

UPES Campus Tel : +91-135-2770137

“Energy Acres” Fax : +91 135- 27760904

P.O. Bidholi, , Dehradun

**COURSE PLAN**

1. **PREREQUISITE:**

a. Basic Knowledge of C-Programming, Data types

b. Basic data structures and oops concepts

1. **PROGRAM OUTCOMES (POs) and PROGRAM SPECIFIC OUTCOMES (PSOs) for CCVT:**

**B1. PROGRAM OUTCOMES (POs)**

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.

**B2. Program Specific Outcomes (PSOs)**

**PSO1:** Perform system and application programming using computer system concepts, concepts of Data Structures, algorithm development, problem solving and optimizing

**PSO2:** Apply software development and project management methodologies using concepts of front-end and back-end development and emerging technologies and platforms.

**PSO3:** Understand and apply Cloud Computing architecture for scalable, secure and dynamically provisioned business oriented environment with optimized performance tuning and data reliability.

1. **COURSE OUTCOMES FOR INTRODUCTION TO VIRTUALIZATION AND CLOUD COMPUTING: At the end of this course student should be able to**
2. Understand the Object Oriented Programming concepts.
3. Familiar with C++ programming.
4. Model the solutions using OOPs methodology.
5. Analyze various data structures and their design techniques.
6. Apply advanced data structure strategies to solve real world problems.
7. Devise new data structures.
8. Implement the solutions for real world applications using C++

**Table: Correlation of POs and PSOs v/s COs**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Engineering Knowledge | Problem analysis | Design/development of solutions | Conduct investigations of complex problems | Modern tool usage | The engineer and society | Environment and sustainability | Ethics | Individual or team work | Communication | Project management and finance | Life-long Learning | Perform system and application programming using computer system concepts | Apply software development and project management methodologies | Ability to understand and apply Cloud Computing architecture for scalable, secure and dynamically provisioned business oriented environment with optimized performance tuning and data reliability. |
| Course Code | Course Title | PO1 | PO2 | PO3 | PO 4 | PO 5 | PO6 | PO 7 | PO8 | PO9 | PO 10 | PO 11 | PO12 | PSO1 | PSO2 | PSO3 |
| CSEG-1004 | Advanced Data structure | 1 | 2 | 1 | 2 |  |  |  |  |  |  | 1 |  | **3** | **2** | **2** |

1=weakly mapped 2= moderately mapped 3=strongly mapped

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping between COs and POs** | | |
|  | **Course Outcomes (COs)** | **Mapped Programmed Outcomes/PSO** |
| **CO1** | Understand the Object Oriented Programming concepts. | **PO1, PSO 3** |
| **CO2** | Familiar with C++ programming. | **PO 3, PO 5** |
| **CO3** | Model the solutions using OOPs methodology. | **PO 2, PSO 1** |
| **CO4** | Analyze various data structures and their design techniques | **PO 3,PO4, PSO1, PSO3** |
| **CO5** | Apply advanced data structure strategies to solve real world problems. | **PO4, PSO2,PSO3** |
| **CO6** | Devise new data structures. | **PO1, PSO1** |
| **CO7** | Implement the solutions for real world applications using C++ | **PO11,PSO2,PSO3** |

**Table: Correlation of POs v/s COs**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PO/CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 | PSO3 |
| CO1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  | 2 |
| CO2 |  |  | 1 |  | 1 |  |  |  |  |  |  |  |  |  |  |
| CO3 |  | 2 |  |  |  |  |  |  |  |  |  |  | 2 |  |  |
| CO4 |  |  | 1 | 1 |  |  |  |  |  |  |  |  | 2 |  | 3 |
| CO5 |  |  |  | 1 |  |  |  |  |  |  |  |  |  | 2 | 2 |
| CO6 | 1 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |
| CO7 |  |  |  |  |  |  |  |  |  |  | 1 |  |  | 2 | 2 |

1. WEAK 2. MODERATE 3. STRONG

1. **COURSE OUTLINE**

The course provides the deep understanding of advanced data structures (Tree and Graphs) and their implementation in C++ programming language

|  |  |  |
| --- | --- | --- |
| **S.No** | **Unit** | **Contents** |
| 1 | Unit - 1 | Introduction |
| 2 | Unit – 2 | Inheritance & Polymorphism |
| 3 | Unit – 3 | File handling and Hashing |
| 4 | Unit – 4 | Tree Data structure |
| 5 | Unit – 5 | Advanced Tree Data structure |
| 6 | Unit - 6 | Graph Data Structures |

.

1. **PEDAGOGY**
2. Class Test
3. Quiz
4. Assignment
5. Digital Presentations
6. Reflection
7. **COURSE COMPLETION PLAN**

|  |  |
| --- | --- |
| **Total Class room Sessions** | 48 |
|  |  |
| **Total Test** | 02 |
| **Total Assignments** | 02 |
| **Total Reflections** | 04 |

One Session=60 min

Students will be evaluated based on the following 3 stages.

Internal Assessment - 30%

Mid-term Examination - 20%

End term Examination - 50%

**G. INTERNAL ASSESSMENT: WEIGHTAGE – 30%**

Internal Assessment shall be done based on the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Description** | **Marks** | **% of Weightage out of 30%** |
| 1 | Test | 2 online Tests @ 15 Marks each | 30 |
|  |  |  |  |
| 2 | Assignment | Two Online Assignments @ 12 Marks each | 24 |
| 4 | Reflection | 4 Online Reflections @ 9 Marks each | 36 |
| 5 | Attendance | @10 Marks 10% | 10 |
| **Total** | | | **100%** |

**G1*.*** Internal Assessment Record Sheet (including Mid Term Examination marks) will be shown to the students at the end of semester i.e. last week of regular classroom teaching.

**G2. CLASS TESTS/QUIZZES:** One Class Test based on descriptive type theoretical & numerical questions and Two Quizzes based on objective type questions will held. Those who do not appear in Viva-Voce and quiz examinations shall lose their marks.

**G3. ASSIGNMENTS:** There will be four assignments based on theory and numerical problems. Two assignment will be given before mid-semester and two after mid semester examination. Those who fail to submit the assignments by the due date shall lose their marks.

**G4. REFLECTION:** Five reflections will be uploaded on Black board.

**G5. GENERAL DISCIPLINE:** Based on student’s regularity, punctuality, sincerity and behavior in the class.

**G6. MID TERM EXAMINATION: WEIGHTAGE – 20%**

Mid Term examination shall be online which includes the objective type of questions.

**G7. END TERM EXAMINATION: WEIGHTAGE – 50%**

End-Term examination shall be three hours duration and shall be a combination of short, long and very long theory/numerical questions.

**G8. GRADING:**

The overall marks obtained at the end of the semester comprising all the above three mentioned shall be converted to a grade.

1. **COURSE DELIVERY PLAN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SESSION** | **TOPIC** | **Course Outcomes Addressed/Sub Objectives** | **Required Learning Resources**  **(including media)** | **Discussion(s) and  Postings on Frontier** | **Assignment(s)/Quizzes/ Tests** |
| **1** | Bridging the concepts from procedural languages to object oriented languages: Structures, Classes, Design a comparative program | **CO1** | **Text Book** | **Lecture and Discussion** |  |
| **2** | Object oriented programming paradigm, Data Types in C++, Functions, Parameter Passing Sample Programs | **CO1,CO2** | **Text Book** | **Lecture and Discussion** |  |
| 3 | Basic concept of object oriented programming, running programs under Linux platform, Benefits of OOPS, Object oriented languages and applications of OOP. | **CO2** | **Text Book** | **Lecture and Discussion** |  |
| 4 | Class definitions, Objects, Class Members, Access Control, Parameter Passing methods(Online) | **CO2** | **Text Book** | **HBO** |  |
| 5 | Constructors & Types of Constructor-I | **CO2** | **Text Book** | **Lecture and Discussion** |  |
| **6** | Types of Constructor-II, Destructor | **CO2** | **Text Book** | **Lecture and Discussion** |  |
| **7** | Inline, Static Members, This Pointer | **CO2** | **Text Book** |  |  |
| **8** | Friend Functions, friend classes\ | **CO2** | **Text Book** | **Lecture and Discussion** |  |
| **9** | Dynamic memory allocation and De-allocation, new and Delete | **CO2** | **Text Book** | **Lecture and Discussion** | **Reflection -1** |
| **10** | Inheritance Concepts and definitions, Types and Modes-Basics | **CO2** | **Test Book** | **Lecture and Discussion** |  |
| **11** | Base Class Access Control depending on modes of inheritance | **CO3** | **Test Book** | **Lecture and Discussion** |  |
| **12** | Polymorphism - Static Concepts, Function Overloading, Default arguments(Online) | **CO3** | **Text Book** | **HBO** |  |
| **13** | Operator overloading without friend functions | **CO2** | **Text Book** | **Lecture and Discussion** |  |
| **14** | Operator overloading – with friend functions-I | **CO3** | **Test Book** | **Online via** |  |
| **15** | Operator overloading – with friend functions-II | **CO3** | **Text Book** | **HBO** | **Assignment-1** |
| **16** | Dynamic polymorphism, Virtual Functions-I | **CO3** | **Text Book** | **Lecture and Discussion** |  |
| **17** | Dynamic polymorphism, Virtual Functions-II(Online) | **CO3** | **Text Book** | **HBO** |  |
| **18** | Abstract Classes, Ambiguity Resolution | **CO3** | **Text Book** | **Lecture and Discussion** | **Reflection -2** |
| **19** | Generic Functions, Generic Classes-I | **CO3** | **Text Book** | **Lecture and Discussion** |  |
| **20** | Generic Functions, Generic Classes-II(Online) | **CO3** | **Text Book** | **HBO** | **Test-1** |
| **21** | Introduction to Exception Handling | CO3 | **Text Book** | **Lecture and Discussion** |  |
| **22** | Exception Handling-I | **CO3** | **Text Book** | **Lecture and Discussion** |  |
| **23** | Exception Handling-II | **CO3** | **Test Book** | **Lecture and Discussion** |  |
| **24** | File Handling, Streams and Modes, Opening and Closing A file | **CO3** | **Text Book** | **Lecture and Discussion** |  |
| **25** | File Handling: File Position Pointers(Online) | **CO2** | **Test Book** | **HBO** |  |
| **26** | File Handling: Random access in a file | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| **27** | Hash Tables: Direct Address tables, hash functions: collision resolution-separate chaining | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| 28 | open addressing-linear probing, quadratic probing | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| 29 | double hashing, rehashing, extendible hashing(Online) | **CO4** | **Text Book** | **HBO** | **Reflection -3** |
| **30** | Introduction to Data Structures | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| **31** | Binary tree, terminology | **CO3** | **Text Book** | **HBO** |  |
| 32 | complete binary tree, extend binary tree | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| **33** | traversing binary tree, threaded binary tree(Online) | **CO4** | **Test Book** | **HBO** |  |
| **34** | Binary search tree & operations - Searching, Insertion | **CO4** | **Test Book** | **Lecture and Discussion** |  |
| **35** | Binary search tree | **CO4** | **Text Book** | **Lecture and Discussion** |  |
| **36** | BST operations - Deletion-II | CO4 | **Text Book** | **Lecture and Discussion** |  |
| **37** | AVL Trees, Operations- Insertion(Online) | **CO4** | **Text Book and Reference book** | **HBO** |  |
| **38** | AVL Trees, Operations-Deletion |  | **Text Book and Reference book** | **Lecture and Discussion** |  |
| **39** | m-way search tree, operations-, insertion | **CO5** | **Text Book and Reference book** | **Lecture and Discussion** |  |
| **40** | m-way search tree operations-deletion | **CO5** | **Text Book and Reference book** | **Lecture and Discussion** |  |
| **41** | m-way search tree operations-searching (Online) | **CO5** | **Text Book and Reference book** | **HBO** |  |
| **42** | heaps, maintaining heaps property | **CO5,CO7** | **Test Book** | **Lecture and Discussion** |  |
| **43** | heap sort, priority queue | **CO5,CO7** | **Test Book** | **Lecture and Discussion** | **Assignment-2** |
| **44** | Graph-terminology, multigraph, digraph | **CO7** | **Test Book** | **Lecture and Discussion** |  |
| **45** | sequential representation, linked list representation | **CO7** | **Test Book** | **Lecture and Discussion** |  |
| **46** | graph traversals (DFS & BFS)(Online) | **CO6** | **Test Book** | **HBO** |  |
| **47** | path matrix, completely connected graph | **CO6** | **Test Book** | **Lecture and Discussion** | **Test-2** |
| **48** | Applications of Graphs | **CO6** | **Test Book** | **Lecture and Discussion** | **Reflection -4** |

1. **SUGGESTED READINGS:**

**I.1 Text 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.

**I.2. Reference Books**

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

**GUIDELINES**

***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.

***E-Mail and online learning tool:*** Each student in the class should have an e-mail id and a password 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.

***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.

**Course outcome assessment:** To assess the fulfilment of course outcomes two different approaches have been decided. Degree of fulfillment of course outcomes will be assessed in different ways through direct assessment and indirect assessment. In Direct Assessment, it is measured through quizzes, tests, assignment, Mid-term and/or End-term examinations. It is suggested that each examination is designed in such a way that it can address one or two outcomes (depending upon the course completion). Indirect assessment is done through the student survey which needs to be designed by the faculty (sample format is given below) and it shall be conducted towards the end of course completion. The evaluation of the achievement of the Course Outcomes shall be done by analyzing the inputs received through Direct and Indirect Assessments and then corrective actions suggested for further improvement. Capping

***Passing criterion:*** Student has to secure minimum 40% marks of the “highest marks in the class scored by a student in that subject (in that class/group class)” individually in both the ‘End-Semester examination’ and ‘Total Marks’ in order to pass in that paper.

* Passing Criterion for B. Tech: Minimum 40% of the highest marks in the class.
* Passing Criterion for M. Tech: minimum 40% of the highest marks in the class

**Sample format for Indirect Assessment of Course outcomes**

|  |
| --- |
| NAME: |
| ENROLLMENT NO: |
| SAP ID: |
| COURSE: |
| PROGRAM: |

Please rate the following aspects of course outcomes of B.Tech(CSE), 3rd sem, Advanced Data Structures. Use the scale 1-4\*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sl. No. | Course outcomes | 1 | 2 | 3 | 4 |
| 1 | Understand the Object Oriented Programming concepts. |  | Y |  |  |
| 2 | Familiar with C++ programming. |  |  | Y |  |
| 3 | Model the solutions using OOPs methodology. |  | Y |  |  |
| 4 | Analyze various data structures and their design techniques. |  |  | Y |  |
| 5 | Apply advanced data structure strategies to solve real world problems. |  |  |  | Y |
| 6 | Devise new data structures. |  |  |  | Y |
| 7 | Implement the solutions for real world applications using C++ |  |  | Y |  |

3

Below Average

Good

1

**\***

Very Good

Average

4

2