****

**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 spl. in Cloud Computing &Virtualization Technologies

Course :Programming and Data Structures

Subject Code : CSEG1002

No. of credits : 4

Semester : I

Session : July 2018 – December 2018

Batch : 2018 - 2022

Prepared by : Ms. Ambika Aggarwal

Email : ambika.aggarwal@ddn.upes.ac.in

**Approved By**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Faculty HOD

UPES Campus Tel: +91-135-2770137

“Energy Acres” Fax: +91 135- 27760904

P.O. Bidholi, Via Prem Nagar, Dehradun Website: [www.upes.ac.in](http://www.upes.ac.in)

**COURSE PLAN**

1. **PRE-REQUISITES**

Elementary knowledge of computers.

1. **PROGRAM OUTCOMES (POs) and PROGRAM SPECIFIC OUTCOMES for B.Tech. CSE spl. in Graphics and Gaming**

**B1. PROGRAM OUTCOMES (POs)**

1. *Engineering knowledge:* Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. *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.
3. *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.
4. *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.
5. *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.
6. *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.
7. *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.
8. *Ethics:* Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. *Individual and team-work:* Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. *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.
11. *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.
12. *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)**

1. Perform system and application programming using computer system concepts, concepts of Data Structures, algorithm development, problem solving and optimizing techniques.
2. Apply software development and project management methodologies using concepts of front-end and back-end development and emerging technologies and platforms.
3. Understand and apply Cloud Computing architecture for scalable, secure and dynamically provisioned business oriented environment with optimized performance tuning and data reliability.
4. **COURSE OBJECTIVES**
5. To help the students to learn the basics of C programming language.
6. To enable students to develop programming skills.
7. To give the students basic and intermediate knowledge about the C programming language and data structure.
8. To enable students to develop programs of intermediate level in C programming language.
9. To make students aware about the concept of data structures.
10. To enable students to implement the concept of data structure using C programming language.
11. **COURSE OUTCOMES (COs), Mapping with POs and PSOs**

Upon completion of this course the learners will be able to:

1. Demonstrate a clear understanding of the basic C programming concepts.
2. Use functions, storage class specifiers, pointers and dynamic memory allocation.
3. Implement the basic data structures like arrays, structures, linked lists, stacks and queues.
4. Classify and analyze the complexities associated with sorting/searching algorithms and demonstrate a clear understanding of the file handling concepts.
5. Develop skills to design and evaluate solutions to real time problems.

**Table: Correlation of COs vs. POs for the Course**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PO/PSO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 | PSO3 |
| CO1 | 1 |  |  |  |  |  |  |  | 2 |  |  |  |  |  |  |
| CO2 |  |  | 2 |  |  |  |  |  |  |  |  |  | 2 |  |  |
| CO3 |  |  | 2 |  |  |  |  |  |  |  |  |  | 2 |  |  |
| CO4 |  | 1 |  | 2 |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  |
| **CSEG1002** | **1** | **1** | **3** | **2** |  |  |  |  | **2** |  |  |  | **2** |  |  |

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

1. **COURSE OUTLINE**

|  |  |
| --- | --- |
| **Module** | **Contents** |
| 1 | Program Development Steps, Basic Concepts in C |
| 2 | Pointers, Functions, Storage Class Specifiers |
| 3 | Arrays, Dynamic Memory Allocation, Structure, Union |
| 4 | Data Structures (Stack, Queue, Linked List) |
| 5 | Searching, Sorting, File Handling |

1. **PEDAGOGY**

* Presentations
* Flipped Classroom sessions
* Think-Pair-Share Activities
* Video Lectures

1. **COURSE COMPLETION PLAN**

|  |  |
| --- | --- |
| **Total Sessions** | 48 (F2F: 36, Online: 12) |
| **Total Continuity Assessments** | 5 |
| **Total Tests** | 2 |
| **Total Assignments/Project** | 2/1 |

One Session = 60 minutes

1. **EVALUATION & GRADING**

The components of the instructor-led continuous evaluation system will be as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Assessment** | **Weightage** | **Schedule** |
| 1 | Internal Assessment (IA) | 30% | Detailed Below |
| 2 | Mid-semester Examination (MS) | 20% | Academic Calendar |
| 3 | End-semester Examination (ES) | 50% | Academic Calendar |
| Total | | 100% |  |

Each Assessment is carried out for suitable marks and finally reduced suitably based on its weightage. At course completion, the student is awarded with the grade based on the composite score obtained out of 100 marks (30% IA + 20% MS + 50% ES). While awarding the grades, the evaluator will necessarily award grade “F” if the raw score obtained by the student is less than 40%of the highest raw score obtained in ES or/and in the composite score (IA + MS + ES). Refer the student bulletin in the intranet for any additional information. It is mandatory for all the students to undergo the process of continuous evaluation.

1. **Internal Assessment:**

Internal Assessment shall be done based on the following detailed breakup and scheme of assessment:

|  |  |  |
| --- | --- | --- |
| **Assessment** | **Points** | **Percentage** |
| Continuity Assessment (CA) | 5 online CA’s after/nearing completion of every Module @10 points each | 50% |
| Assignments **or** Project | 1 project @20 points **(OR)**  2 Assignments @10 points each | 20% |
| Test | 2 online Tests after/towards the end of Module 2 & 5 @10 points each | 20% |
| Conduct of the student | Participation in Online and F2F lessons @10 points | 10% |
| **Total** | **100 points** | **100%** |

The marks awarded for the Online Internal Assessments will be available in Black Board and displayed to the students.

1. **Mid-semester Examination:**

Mid-semester examination will cover approximately half of the entire course content and shall be of two hours duration. The question paper pattern would be discussed well in advance before the exam. The evaluated answer sheets of the written exam shall be disclosed to the students ten days after the examinations.

1. **End-semester Examination:**

End-semester examination will cover the entire course content and shall be of three hours duration. The examination shall have short answer type questions, analytical and conceptual comprehension through essay/descriptive type questions, and cases or problem solving exercises. The evaluated answer sheets shall be disclosed to the students ten days after the examinations.

**GRADING:**

The overall marks obtained at the end of the semester comprising the above three shall be converted to a grade. Student(s), who have met the qualifying criteria of the individual theoretical subject but have not met qualifying criteria of SGPA, will not be allowed to re-appear for improvement. Students, who wish to re-appear in the theoretical subject, shall be required to pay the prescribed fee per subject as notified by the University.

The student with Grade “F” will be eligible to repeat continuous evaluation of that respective subject (s) during summer vacation (June-July). Grade shall be awarded on the performance of the student(s). The Grade will be capped as per the rules mentioned in student Bulletin. All other rules and regulations such as requirement of passing, etc. will remain same as mentioned in rules & regulations.

1. **DETAILED SESSION PLAN**

Note: The Online Sessions are highlighted in yellow color.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Module /Session** | **Big Ideas/ Topics** | **Course Outcomes Addressed** | **Required Learning Resources (including media)** | **Pedagogy/ Discussion(s)/ Postings** | **Assessment** |
| **Module 1** | **Basic C Programming Concepts** |  |  |  |  |
| **1** | Computer Fundamentals, Algorithm, Pseudo code and Flow-chart | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **2** | Program Development steps, Use of translators, Linkers, Loaders, Editors and Locaters | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **3** | Structure of C program, A Simple C program, Identifiers, Data types, Sizes of Data Types | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **4** | Size-of operator, Modifiers, Use of values.h and limits.h | CO1 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Project Release\* |
| **5** | Variables, Declaration vs. Definition, Types of Variables, Global vs. Local Variable, Literals, Constants & Qualifiers | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **6** | Arithmetic operators, Relational and Logical operators | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **7** | Assignment operators, Increment and Decrement operators, Conditional operator, Bit-wise operators | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **8** | Expressions, Precedence and Order of evaluation (Associativity), Type conversions | CO1 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation |  |
| **9** | if statements, Conditional expressions, Input-output statements, statements and blocks | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **10** | Loops: while, do-while, for statements | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **11** | Switch statements, break, continue, goto and labels | CO1 | Refer Blackboard, Suggested Reading | Lecture |  |
| **12** | Online Discussion & Continuity Assessment | CO5 | Refer Blackboard, Suggested Reading | Readings/ web search | Continuity Assessment -1 |
| **Module 2** | **Pointers and Functions** |  |  |  |  |
| **13** | Pointers - concepts, initialization of pointer variables, Pointers to pointers | CO2 | Refer Blackboard, Suggested Reading | Lecture |  |
| **14** | Designing structured programs, Function basics | CO2 | Refer Blackboard, Suggested Reading | Lecture |  |
| **15** | Standard library functions, Creating and Linking our own files. | CO2 | Refer Blackboard, Suggested Reading | Lecture |  |
| **16** | Parameter passing (Function Call by Value, Call by pointers), Function Pointers | CO2 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Assignment-1 Release |
| **17** | Command Line Arguments, Recursive function, Variadic Functions | CO2 | Refer Blackboard, Suggested Reading | Lecture |  |
| **18** | Storage classes - extern, auto, register, static, scope rules | CO2 | Refer Blackboard, Suggested Reading | Lecture |  |
| **19** | Re-visiting | CO1, CO2 | Refer Blackboard, Suggested Reading | Discussion |  |
| **20** | Online Discussion & Continuity Assessment | CO5 | Refer Blackboard, Suggested Reading | Readings/ web search | Continuity Assessment-2 |
| **Module 3** | **Array, Structure and Union** |  |  |  |  |
| **21** | Introduction to data structures, Arrays - concepts, declaration, definition, One-Dimensional arrays, Storage representation, address arithmetic, character array (introduction to strings), Character pointers | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **22** | Two-dimensional and Multi-dimensional arrays | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **23** | Variable length arrays, Arrays and functions (Array as functions argument) | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **24** | Dynamic memory managements functions, string handling using library functions | CO3 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Assignment-1 Submission |
|  | Internal Assessment Test |  |  |  | Test-1 (Covers Modules 1&2) |
| **25** | Derived types - Structures- declaration & definition, Memory allocation | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **26** | Arrays of structures, Structures and functions | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **27** | Pointers to structures - Access structure members using pointers, Self-referential structures, Nested structures, | CO2, CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **28** | Unions, typedef (creating structures and union), ADT | CO3 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation |  |
|  | Online Discussion & Internal Assessment | CO5 |  |  | Continuity Assessment-3 |
| **Module 4** | **Queue, Stack and Linked List** |  |  |  |  |
| **29** | Representing Queues using Arrays | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **30** | Representing Stacks using Arrays | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **31** | Polish and Reverse polish Notations, Conversion from Infix to Polish and Infix to Reverse Polish notations | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **32** | Postfix and Prefix expression evaluation, Check for Balanced Parenthesis, Reverse String using Stack | CO5 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Assignment-2 Release |
| **33** | Linked lists: Operations (insert) | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **34** | Linked lists: Operations (delete & search) | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **35** | Stack & Queue implementation using Linked List | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **36** | Circular Queues, Priority Queues, Circular Linked List | CO5 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation |  |
| **37** | Doubly linked lists | CO3 | Refer Blackboard, Suggested Reading | Lecture |  |
| **38** | Polynomial representation & operations using Linked List | CO5 | Refer Blackboard, Suggested Reading | Lecture |  |
| **39** | Re-visiting | CO1-CO3 | Refer Blackboard, Suggested Reading | Discussion |  |
| **40** | Online Discussion & Continuity Assessment | CO5 | Refer Blackboard, Suggested Reading | Readings/ web search | Continuity Assessment-4 |
|  | Project | CO5 |  |  | Project\* Submission (Evaluation starts) |
| **Module 5** | **Searching, Sorting and File Handling** |  |  |  |  |
| **41** | Complexity of Algorithms, Searching - Linear and Binary search methods | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **42** | Sorting - Bubble sort, Selection sort | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **43** | Insertion sort, Merge Sort | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **44** | Quick sort, Comparative Study of the different Sorting and Searching Algorithms, Comparison of Different Data Structures. | CO4 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Assignment-2 Submission |
|  | Online Internal Assessment Test |  |  |  | Test-2 (Covering Modules 3,4 &5) |
| **45** | File Handling: Text Files vs Binary Files, Modes of operation (Input and Output) | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **46** | Handling Binary files | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **47** | Streams, Standard I/O, Formatted I/O | CO4 | Refer Blackboard, Suggested Reading | Lecture |  |
| **48** | Random File I/O operations | CO4 | Refer Blackboard, Suggested Reading | Readings/ brief video/ Presentation | Project\* Evaluation ends |
|  | Online Discussion & Internal Assessment | CO5 |  |  | Continuity Assessment-5 |

\*The instructor can either opt for a project or assignments. However, project is preferable.

1. **SUGGESTED READING**

**J.1 Text Books**

1. Seymour Lipschutz, “Data Structures with C (Schaum's Outline Series)”.
2. Yashavant P. Kanetkar, “Let us C”.

**J.2 Reference Books**

1. E. Balagurusamy, “Programming in ANSI C”.
2. Ellis Horowitz and Sartaz Sahni, “Data Structure using C”.
3. P.K. Sinha and P.K. Sinha, “Computer Fundamentals”.

Note: Also refer to the Web-links/Resources in Blackboard and NPTEL videos.

1. **GUIDELINES**

**Cell Phones and other Electronic Communication Devices:** Cell phones and other electronic communication devices (such as Blackberries/Laptops) must be turned off during the lab session.

**e-Mail and online learning tool:** Each student in the class should have UPES e-mail id and a password to access the Blackboard regularly. The best way to arrange meetings with faculty is by email and prior appointment. 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 the subject.

**Passing criterion:** Student has to secure minimum 40% marks of the “highest marks scored by the student for the subject” in the total marks in order to pass in that paper.

1. **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 Continuous assessments. Each assessment 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.

**Format for Indirect Assessment of Course Outcomes**

|  |  |
| --- | --- |
| NAME: |  |
| ENROLLMENT NO: |  |
| SAP ID: |  |
| COURSE: | Programming and Data Structures |
| PROGRAMME: |  |

Please rate the following aspects of the Course Outcomes. Use the scale 1 to 4 \*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S. No.** | **Course Outcomes** | **1** | **2** | **3** | **4** |
| CO1 | Demonstrate a clear understanding of the basic C programming concepts. |  |  |  |  |
| CO2 | Use functions, storage class specifiers, pointers and dynamic memory allocation. |  |  |  |  |
| CO3 | Implement the basic data structures like arrays, structures, linked lists, stacks and queues. |  |  |  |  |
| CO4 | Classify and analyze the complexities associated with sorting/searching algorithms and demonstrate a clear understanding of the file handling concepts. |  |  |  |  |
| CO5 | Develop skills to design and evaluate solutions to real time problems. |  |  |  |  |

Very Good

4

3

Below Average

Average

Good

2

1

\*