Department of Computer Science and Engineering, Indian Institute of Technology Madras

<table>
<tr><td>Course title</td><td colspan="9">Problem Solving using Computers</td><td>Course No</td><td colspan="4">CS1111</td></tr>
<tr><td rowspan="2">Department</td><td rowspan="2">Computer Science and Engineering</td><td rowspan="2">New Credits</td><td>L</td><td>T</td><td>E</td><td>P</td><td>O</td><td>C</td><td>TH</td><td rowspan="2">Old Credits</td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td>3</td><td></td><td></td><td>3</td><td>6</td><td>12</td><td>12</td><td>3</td><td></td><td>1</td><td>4</td></tr>
<tr><td>Offered for</td><td colspan="9"></td><td>Status</td><td colspan="4">Modified</td></tr>
<tr><td>Faculty</td><td colspan="9">Rupesh Nasre</td><td>Type</td><td colspan="4">Theory</td></tr>
<tr><td>Pre-requisite skills</td><td colspan="9">NIL</td><td>To take effect from</td><td colspan="4"></td></tr>
<tr><td>Submission date</td><td colspan="3">Date of approval by DCC</td><td colspan="6">Date of approval by BAC</td><td colspan="5">Date of approval by Senate</td></tr>
<tr><td></td><td colspan="3"></td><td colspan="6"></td><td colspan="5"></td></tr>
</table>

Objectives:
The course aims to provide exposure to problem-solving through programming. It aims to train the student to the basic concepts of the C programming language. This course involves a lab component which is designed to give the student hands-on experience with the concepts.

Course Contents:
The course is structured around 8--10 problems that will be solved throughout the semester. For each problem the necessary language constructs will be introduced.

Weeks 1 and 2: Introduction and getting started, Motivation of the course, fundamentals of computing, historical perspective, writing the first C program. Phases of developing and running a program in C. Statements, declarations, variables, basic data types (int, char), basic arithmetic and logical operators, if-else and loop construct.

Week 3: Input from user (scanf), output to screen (printf), Arithmetic operators, precedence of operators. Sample Problem: Weighted sum of assignment marks for a single student.

Week 4: Introduction to arrays, use of loops, need for different loop structures like while, do-while. Sample Problem: Weighted sum of assignment marks for several students.

Week 5: Introduction to character arrays, manipulating character arrays, motivate need for functions. Sample Problem: Encrypt and Decrypt a string.

Week 6: 2-D arrays, use of loops, user input, input validation. Sample Problem: Tic-Tac-Toe program (2 users)

Week 7: Functions. The prototype declaration, Function definition. Function call - Passing arguments to a function, by value, by reference. Scope of variable names.

Week 8: Recursion, recursive formulas, factorial, Hemachandra (aka Fibonacci) numbers, recursive functions in C, sum, max of integers in an array, GCD program. Some other example problems are : Towers of Hanoi, Binomial coefficients.

Week 9: Need to dynamically allocate memory, pointers, introduction, integer pointers, character pointers, array of pointers. Sample Problem: Encrypt and Decrypt multiple strings of variable lengths

Week 10: Aggregate data types, structures, unions, structures and functions, use of .h file, multi-file compilation, makefiles. Sample Problem: Maintaining Student records

Week 11: Array of structures, linear search for a key value, searching in sorted arrays, binary search, sorting algorithms - bubble sort, insertion sort. Sample Problem: Searching and Sorting Student records

Week 12: Self Referential Structures: Introduction to Linked lists, integer linked lists, insert, delete, search in linked list. Recursion and linked list.

Week 13: Linked list (queue), insertion, removal. Sample Problem: Developing a token system at bank. Can be enhanced to support multiple service windows in the bank, so deletions may happen for an internal node in the list.

Week 14: File handling, reading, and writing to files, command line parameters. Sample Problem: Ticket booking system. A simpler version of IRCTC ticket booking for one train, one bogie. Bookings stored in a file, which should be readable on multiple executions of a.out.

Text Books:
C: How to program, H. M. Deitel, P. J. Deitel, 7th edition, Pearson Education, 2010.

Reference Books:
R. G. Dromey, "How to Solve It By Computer", Pearson, 1982A.R. Bradley, "Programming for Engineers", Springer, 2011A.G. Ranade, "An Introduction to Programming through C++", McGraw Hill, 2014Kernighan and Ritchie, "The C Programming Language", (2nd ed.) Prentice-Hall, 1988