Course number and name: CS 22900  C/C++ Programming for Electrical and Computer Engineering (Short Title, C/C++ Programming)

Credits and contact hours: 4.0 Credits; 3.0 Lectures and 1.0 Lab

Instructor's or course coordinator's name: Beomjin Kim

Text book, title, author, and year

Supplemental materials

Course Description
This course will introduce the programming in C and fundamentals of object-oriented programming in C++ to solve engineering problems. This course introduces the C programming language features in the beginning of the semester. Topics covered include data types, control structures, standard input/output, file input/output, mathematical library, problem-solving, functions, arrays, dynamic memory, and pointers. Then the course provides an introduction to object-oriented programming using C++ language. Students will gain understanding of data abstractions, classes, inheritance, composition, method overloading and overriding, generic programming, and standard template library. Students are expected to complete lab assignments and several programming assignments. Prerequisites: ENGR 128 or consent of instructor
Whether a required, elective, or selected elective course in the program: For ECE students only.

Course Goals:
Course Learning Outcomes
Upon successful completion of the course requirements, a student should be able to:

1. The course has improved my ability to use C programming language to solve elementary engineering problems. (a, c, e, k)
2. The course has improved my understanding of and an ability to use data types, variables, and arithmetic operators. (a, e)
3. The course has improved my ability to use conditional statements and loops structures. (c, e, k)
4. The course has improved my understanding of the use of arrays and pointers. (e, k)
5. The course has improved my ability to use the dynamic memory. (c, k)
6. The course has improved my ability to use mathematic library. (c, k)
7. The course has improved my ability to develop function-oriented programs. (c, e, k)
8. The course has improved my understanding of the distinction for passing arguments to/from functions. (c, k)
9. The course has improved my ability to use standard input/output and file input/output operations. (e, k)
10. The course has improved my ability to use object oriented programming in C++ to solve basic engineering problems. (a, c, e, k)
11. The course has improved my understanding of the use of classes and access control to class members. (c, k)
12. The course has improved my ability to use class inheritance and composition. (c, k)
13. The course has improved my understanding of the method overloading and overriding. (j, k)
14. The course has improved my ability to use templates and standard template library. (c, k)

**Student Outcomes**

As indicated in the following table, the numbered Course Learning Outcomes (1-14) address the Program's Student Outcomes (a-k).

<table>
<thead>
<tr>
<th>Course Outcome</th>
<th>Program's Student Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
</tr>
<tr>
<td>1</td>
<td>●</td>
</tr>
<tr>
<td>2</td>
<td>●</td>
</tr>
<tr>
<td>3</td>
<td>●</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>●</td>
</tr>
<tr>
<td>11</td>
<td>●</td>
</tr>
<tr>
<td>12</td>
<td>●</td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
</tbody>
</table>

**Major Topics Covered in the Course (Academic Hours)**

- **Introduction to C Language** (3)
  - C program structure,
  - Data types and declarations, Constants and variables, Statements,
  - Using Visual Studio.NET,
  - Preprocessor directives

- **C fundamentals** (6)
  - Arithmetic operators,
  - Standard Input/Output, I/O Formatting,
  - Math Functions,
  - Structured programming, Type casting

- **Decision and Iteration Structures** (6)
  - Conditional expressions, Selection statements, Loop structures

- **File Input/Output, Procedural programming** (3)
Functions and Recursion (6)
   User-defined functions, Arguments passing among functions, Variable scope, Recursion

Arrays and matrices (3)
Problem solving methodology, Pointers (3)
Introduction to C++, OOP, C++ program structure, Classes, Constructor (3)
   OOP, C++ program structure, Classes, Constructor
Access control to class members, Developing OO application, Dynamic memory (3)
Inheritance & composition (3)
Method overloading & overriding (Not covering polymorphism) (3)
Templates and standard template library (3)

Laboratory projects
   At least 8 assignments to cover the course content, all are of one week work

Project assignments
   At least 3 assignments to cover the major topics, all are of several weeks work

How Data in the Course is Used to Assess Program Outcomes
   Each Course Outcome of the assessment survey directly supports one or more of the Program Outcomes (see “Relationship between Course Outcomes and Program Outcomes” above). For CS 22900, Program Outcomes a, c, e, j, and k are supported.