TO: The Faculty of the College of Engineering
FROM: The Faculty of the School of Electrical and Computer Engineering
RE: New Undergraduate Level Course ECE 435

The faculty of the School of Electrical and Computer Engineering has approved the following new course. This action is now submitted to the Engineering Faculty with a recommendation for approval.

ECE 435   Object-Oriented Design Using C++ and Java

Sem. 2, Class 3, cr. 3.
Prerequisite: ECE 462 and Consent of Instructor

Review of OO design with C++ and Java. Difficulties caused by multiple inheritance in C++. Taking advantage of Run-Time Identification in C++. Multi-threading, AWT, and Network Programming in Java. Discussion of Java applets, beans, and servlets. Unified modeling language. Use-case analysis. Constructing conceptual models. System sequence diagrams. "Gang of Four" design patterns. Case studies.

Reason: It is now widely recognized that just knowing OO languages and having access to a library of classes is not sufficient for creating OO designs. This realization has led to the emergence of a "patterns movement" in the OO community. Patterns are the "best practice" designs that have evolved over the years for tackling issues such as how to make objects sharable; how and when to assign responsibilities to objects; how to make the OO design reusable in other similar contexts, etc.

Mark Smith, Head
School of Electrical & Computer Engineering

Supporting Documentation

Level: Undergraduate Level
Course Instructor: Avinash C. Kak

Course Outline:

Topics	*Lectures*
1. Course Introduction	1
2. Software Development Process for Large OO Programs	1
3. Use Cases, Class Diagrams	1
4. Class Diagrams (Advanced Concepts)	1
5. Interaction, Package, State, and Activity Diagrams	3
6. Extending Classes in C++ and Java	5
7. OO Design using Multiple Inheritance in C++	4
8. Design Patterns	6
9. OO for GUI Design with Java , C++, and C (AWT/Swing in Java, Qt in C++, and GNOME/GTK+ in C)	7
10. OO Design using Multithreading	4
11. OO Design in Network Programming	4
12. Design for Database Programming	3
13. OO Design for Web Sevices Programming	2
14. Exams <u>2</u>	<u>2</u>
Total	**44**

Text(s):

1. <u>UML Distilled, Applying the Standard Object Modeling Language</u>, Martin Fowler and Kendall Scott, Addison-Wesley, 1997, ISBN 0-201-32563-2.

2. <u>Java Design Patterns</u>, James Cooper, Addison-Wesley, 2000, ISBN 0-201-48539-7.

Recommended Reference(s):

CoreJava: Volumes I and II, Cay Horstman and Gary Cornell, Sun Microsystems, 1997, ISBN No: 0-13-766965-8.

C++ Programming Language, 3rd edition, B. Stroustrup, Addison-Wesley, 1997, ISBN No: 0-201-88954-4.

Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, et. al., Addison-Wesley, 1994, ISBN No: 0-201-63361-2.

Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

i. a knowledge of the Unified Modeling Language for the conceptual design of object-oriented programs. [3,4;e,k]

ii. an ability to design object-oriented solutions to programming problems using previously developed "best practice" design components. [3,4;e,k]

iii. an understanding of the pros and cons associated with multiple inheritance in C++. [3,4;e,k]

iv. a knowledge of graphics and user interface programming with Java. [3,4;c,e,k]

v. an ability to carry out databases programming in Java. [3,4;e,k]

vi. an understanding of multithreading issues in Java. [3,4;e,k]

Assessment Methods for Course Outcomes: Each of the outcomes will be assessed by giving the students appropriate C++ and Java programming assignments.