

CSE115: INTRODUCTION TO COMPUTER SCIENCE FOR MAJORS I (4 Credits, required)

Catalog Description

Provides the fundamentals of the field to computer science and computer engineering majors, introducing students to algorithm design and implementation in a modern, high-level programming language. Emphasizes problem solving by abstraction. Topics include object-oriented design using a formal modeling language; fundamental object-oriented principles such as classes, objects, interfaces, inheritance and polymorphism; simple event-driven programming; data types; variables; expressions; basic imperative programming techniques, including assignment, input/output, subprograms, parameters, sequencing, selection and iteration; the use of aggregate data structures, such as arrays or more general collections; simple design patterns.

Prerequisites

Prerequisites: None

Corequisites: MTH 141 or equivalent.

Textbooks(s) and/or other required material

Carl Alphonse and Adrienne Decker. From Conceptual Model to Executable Model: An Object Oriented Problem Solving Process [Notes available at Great Lakes Graphics & Printing, located in the UB Commons.]

Deborah S. Ray & Eric J. Ray. (2006). UNIX: Visual Quick Start Guide (Third Edition), Addison-Wesley, (ISBN: 0321442458)

Carlos Valcarcel. (2005). Eclipse 3.0 Kick Start, Sams Publishing. (ISBN: 0672326108)

Course Objectives

The main objectives of this course are to enable students to analyze a computer science problem, design a solution to it, and implement the design.

Topics Covered

Basic UML class diagrams, and the UML class diagram tool

Objects and classes

Visibility of class members

Object communication

Class relationships (Generalization, Realization, Association, Dependency, Composition)

Types, Type casting, Type checking

Graphical User Interfaces

Event-driven programming

Flow of control constructs (sequencing, selection, repetition)

Collections (arrays)

Searching (linear and binary search)

Class / Lab Schedule

Three 50-minute lectures per week
One 110-minute recitation per week

Contribution of course to professional component/criterion 5

Engineering Topics: 4 credits
Engineering Design

Relationship of course to program outcomes

This course is required of all computer engineering students and has a significant relationship with the following program objectives for computer engineering:

- (a) an ability to apply knowledge of mathematics, probability and statistics, computer science and electrical engineering as it applies to the fields of computer software and hardware
- (g) an ability to effectively communicate technical information in speech, presentation, and writing
- (j) a knowledge of contemporary issues

This course has a strong relationship with the following program objectives for computer engineering:

- (e) an ability to identify, formulate, and solve hardware and software computer engineering problems using sound computer engineering principles
- (k) an ability to use the techniques, skills, and modern hardware and software engineering tools necessary for computer engineering practice

Persons who prepared this description and date of preparation

Carl Alphonse/Adrienne Decker, last updated June 2008