

TCSS 143 Master Syllabus

Version: April 2011

(Approved: 27 May 2011)

Catalog Description

Current description:

Develops fundamental concepts and techniques for analysis, design, and implementation of computer programs using an object-oriented language. Includes graphical user interfaces, event-driven programming, recursive techniques, and simple data structures. Prerequisite: a minimum grade of 2.0 in either TCSS 142 or CSE 142.

Proposed description:

Develops fundamental concepts and techniques for analysis, design, and implementation of computer programs using an object-oriented language. Includes recursive techniques and simple data structures. Prerequisite: a minimum grade of 2.0 in either TCSS 142 or CSE 142.

Preconditions

- Develop and implement programs involving the fundamental programming constructs (variables, types, expressions, assignment, simple I/O, conditional and iterative control structures, functions and parameter passing, structured decomposition).
- Develop and implement programs that use each of the following: arrays, objects, strings.

Student Learning Goals (to be added to syllabus handed out to students)

- Implement a low-complexity program (3 or more interacting classes) which includes the use of interfaces and/or abstract classes, and polymorphism given some design guidance.
- Apply object-oriented design concepts such as inheritance, composition, encapsulation, abstraction, method overloading, method overriding, exception handling, and scope appropriately to the implementation of a program.
- Provide formal documentation for a program using Javadoc comments
- Use single and multidimensional arrays and linked lists as well as basic data structures (Lists, Stacks, Queues) in the implementation of a program.
- Use a provided class, given only its API
- Apply recursive techniques to solve problems

CSS Degree Student Learning Outcomes that this course contributes to (to be added to syllabus handed out to students)

- b. an ability to analyze a problem, identify and define the computing requirements appropriate to its solution;
- c. an ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
- i. an ability to use current techniques, skills, and tools necessary for computing practice.

UWT Student Learning Goals that this course contributes to (to be added to syllabus handed out to students)

Inquiry and Critical Thinking

Students will acquire skills and familiarity with modes of inquiry and examination from diverse disciplinary perspectives, enabling them to access, interpret, analyze, quantitatively reason, and synthesize information critically.

Topics covered

- review of basic programming concepts
- review of basic object-oriented programming concepts
- review of single dimensional arrays and introduction to 2D arrays
- inheritance, interfaces, abstract classes, polymorphism
- lists, stacks, queues, sets, and maps
- recursion
- searching and sorting
- linked structures

Additional Information

This course has an associated lab section that meets once per week in addition to the 2 lectures per week.

There is an optional 2 credit seminar workshop (TCSS 390A) associated with this course.

The textbook used in recent years:

Building Java Programs Second Edition, Stuart Reges and Marty Stepp, Addison Wesley, ISBN-10: 0-13-609181-4