COURSE DESCRIPTION

Dept., Number	CSci 581/582	Course Title	Special Topics in Computer Science I/II: Multiparadigm Programming in Scala
Semester hours	3	Course Coordinator	H. Conrad Cunningham, Professor

Current Catalog Description

Special topics in computer science.

Topic description: This study in Fall 2008 examines the principles and practice of multiparadigm programming using the new programming language Scala, a language that executes on the Java platform. In particular, the Scala language directly supports programming in the imperative, object-oriented, and functional paradigms. In addition, Scala's Actor library provides (Erlang-like) concurrency-oriented programming using asynchronous message-passing, and Scala's pattern-matching features, XML and parser combinator libraries, and flexible operator syntax provide facilities for language-oriented programming. As appropriate, the course addresses other topics drawn from areas of contemporary interest such as multicore processor programming, domain-specific languages, software frameworks, and reusable components.

Textbook

Martin Odersky, Lex Spoon, and Bill Venners. *Programming in Scala: A Comprehensive Step-By-Step Guide*, Artima, Inc., to appear July 2008.

Martin Odersky. *Scala by Example*, draft May 2008. http://www.scala-lang.org/docu/files/ScalaByExample.pdf.

References

Scala language website: http://www.scala-lang.org/ .

Course website: http://www.cs.olemiss.edu/~hcc/581scala (under constuction).

Course Outcomes

Upon successful completion of this course, the students:

1. know the fundamental concepts and techniques of the (a) object-oriented, (b) functional, (c) concurrency-oriented, and (d) language-oriented paradigms as embodied in Scala,
2. are able to solve problems by designing appropriate multiparadigm programs using Scala.

Relationship between Course Outcomes and Program Outcomes

This is a course available to beginning computer science graduate students and to advanced undergraduate students as an elective to enrich their programs.

Course outcome 1 contributes to the program outcome (a) and course outcome (2) contributes to program outcomes (b), (c), and (k). The course also contributes to the student's knowledge of a different programming language than most know previously and to their ability to program in different programming paradigms.

Prerequisites by Topic

1. Intermediate programming concepts and skills (CSci 211)
2. Basic data structures and algorithms (CSci 112, 211)
3. Types in programming languages (CSci 112, 211)
4. Recursion (CSci 112, 211)

Major Topics Covered in the Course

This course is under development during Summer and Fall 2008. The major topics are expected to include: introduction to Scala, imperative programming, functional programming, higher-order functions, classes and objects, inheritance and composition, generic typing, traits, lists, "for" comprehensions, mutable objects, concurrency and actors, XML, domain-specific languages, and software frameworks.

Assessment Plan for the Course

This course is under development during Summer and Fall 2008.

How Data in the Course are Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

This course is under development during Summer and Fall 2008.

Estimate Curriculum Category Content (Semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Software design		1
Data structures			Concepts of programming languages		2