COURSE DESCRIPTION

Department and Course Number	CS332F	**Course Coordinator**	Russ Abbott
Course Title	Functional programming	**Total Credits**	2

Current Catalog Description:

Programming in an explicitly functional non-procedural programming language such as Haskell. Programming in a language (a) in which functions are values, (b) without assignments, (c) with very strong typing, (d) with lazy evaluation, and with other features common to the functional programming paradigm

Textbook:

Thompson, Simon., *Haskell: The Craft of Functional Programming, 2nd edition*, Addison Wesley, 1999.

References:

At the discretion of the instructor.

Course Goals:

- To teach students the theory and practice of functional programming.

These course goals contribute to the success of **Student Learning Outcomes 1.a, 1.d, 1.e, 5, and 6**.

Prerequisites by Topic:

Basic competence in a programming language.

Major Topics Covered in the Course:

- Types
- Data structures
- Recursion
- Polymorphism
- Folding
- Functions as values
- Lazy evaluation

Laboratory Projects (specify number of weeks on each):

Typically one each week

Estimate Curriculum Category Content (Quarter Hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms	.25		Data Structures	.25	
Software Design	0.5		Prog. Languages	1.0	
Comp. Arch.					

Oral and Written Communications:

The students are required to complete weekly programming assignments and to present them in class.

Social and Ethical Issues:

No significant component.

Theoretical Content:

The foundations of functional programming.

Problem Analysis:

Students are required to analyze problems whose solutions lend themselves to expression in a functional programming language.

Solution Design:

Students are required to design and develop programs in a functional programming language.