SWE305

Functional Data Structures

TTh, 10 - 11:50, MCT162

Dr. Chen Huo

chuo@cs.ship.edu
Office Location: MCT150
Office Hours:

Course Description:
Immutable data structures make modern software safer and more scalable. In this course, students will learn the functional programming paradigm and how it yields immutable data structures. The data structures covered include lists, trees, and priority queues. Students will learn the operations on the data structures and understand their time complexity. Student will also apply functional solutions to real world problems such as implementing the Game of Life, Tic-Tac-Toe, and solving the Countdown Problem.
Credit Hours: 4

Text(s): *Programming in Haskell*, 2nd Edition
Author(s): Graham Hutton; **ISBN-13:** 9781316626221

Grade Distribution:

Attendance/Participation	10%
Quizzes	60%
Final Exam	30%
Extra Credits	0 – 5%

Letter Grade Distribution:

>= 93.00	A	70.00 - 76.99	C
90.00 - 92.99	A-	60.00 - 69.99	D
87.00 - 89.99	B+	<= 59.99	F
83.00 - 86.99	B		
80.00 - 82.99	B-		
77.00 - 79.99	C+		

Course Objectives:

Functional Programming Basics

1. Understand and use curried functions.
2. Understand and use recursive functions.
3. Understand and use higher order functions.
4. Understand how IO works in a pure functional language.

5. Use type classes to achieve polymorphism.

Data Structures

1. Implement various list operations on an immutable list including folding.
2. Implement the stack data structure.
3. Implement insertion, selection, quick, and merge sort on immutable lists.
4. Implement permutation and combination over a list.
5. Implement tree structures.
6. Implement binary search trees and operations on them.
7. Understand minmax algorithm on a game tree.
8. Implement functional priority queue.
9. Optional: Implement functional red-black tree.
10. Understand the time complexity of operations on covered data structures.

Programming language

1. Understand why immutable data structures are thread-safe.
2. Demonstrate how to use structural induction and equational reasoning to prove computer programs.

Course Policies:

- **Lecture Notes**
  - Slides will be released as notes on D2L after every lecture.
  - Students are expected to take notes. Some contents in the slides and written on the board are not from the textbook. You are recommended to use an actual notebook instead of a computer to take notes. Research show that using a computer to take notes not only distracts other people but also affects your own productivity. You may consider to scan your notes once for a while.
  - All important materials of the course will be managed on D2L, e.g. homework copies, quiz solutions.
- **Homework Submission**
  - Late policy: Homework that is late will be hit with a 20% penalty for the first 48 hours, and will not be accepted thereafter.
  - Format: Homework must be submitted in the correct format (both digital and hard-copy). The wrong format may result in a zero for the certain homework.
- **Grades**
  - All grades that matters will be maintained on D2L. Students are responsible for tracking their progress by referring to the online gradebook. D2L will also tell you what final grade is expected if you are consistent in your performance.
  - You can dispute your grade for a certain item within **ONE** week after the grade is released. For example, you *cannot* dispute your grade of the first homework at the end of the semester.
  - The lowest quiz will be dropped.

- **Attendance**
  - Attendance is expected. Attendance checks are performed in the form of quizzes.
  - If possible, please inform me of your absence before class.
  - If you are not able to make the notice before class, you will have to provide formal evidence, such as doctor's notes, military training notice, etc.

Academic Honesty Policy Summary:
Those students found violating Shippensburg's academic dishonesty policy will be dealt with on a case by case basis. Minimal punishment should it be a first offense is a zero for the assignment and signing a form admitting to the offense. Second offenses are handled directly by the office for the Dean of Students. See additional information about academic Dishonesty within the Computer Science Department

Tentative Course Outline:
The weekly coverage might change as it depends on the progress of the class. The students will be advised when to read the corresponding material.

Week	**Lecture**	**Reading Assignment**
Week 1	First Steps	Ch1, 2
Week 2	Types and Classes	Ch3
Week 3	Functions	Ch4.1 - 4.4
Week 4	Functions	Ch4.5 - 4.8
Week 5	List Comprehensions	Ch5
Week 6	Recursion	Ch6.1 - 6.3
Week 7	Recursion	Ch6.4 - 6.7
Week 8	**Spring Break**	
Week 9	Higher-order Functions	Ch7
Week 10	More about Lists	Notes
Week 11	Trees, Heaps	Notes
Week 12	Trees, Heaps	Notes
Week 13	Interactive Programs	Ch10
Week 14	TicTacToe	Ch11
Week 15	Reasoning about Programs	Ch16
Final	Final Exam	