

CS 210 – Fact Sheet/FAQ for Doctoral Prelim Exam

- What topics/materials are covered in the advanced software paradigms core area?
 - To successfully pass this area exam, a student is expected to know different software paradigms.
 - Topics in Advanced software paradigms include software architecture styles, component-based architecture, domain-specific architecture (DSSA), design patterns, frameworks, programming language paradigms including concurrent programming languages.
 - The programming language paradigms typically emphasized in this course are: imperative, functional, object-oriented, logical. Others may be mentioned during the lectures.

- How should I prepare for this area and what books should I read?
 - Preparing for the advanced software paradigms area implies reviewing your undergraduate programming language paradigms as well as material from graduate courses in advanced software paradigms.
 - Since the exam may cover a number of topics, it is essential that you cover all the fundamental concepts in this area. The lecture notes posted to the website by Professors Bellaachia, Lancaster, and Kaisler provide a good outline of the fundamental concepts. This is supplemented by required readings, additional material, and suggested references.
 - There are several books that cover the topics mentioned above. As a starting point (i.e., the minimum reading requirement) you may reference the material from the websites of Professors Bellaachia, Lancaster, and Kaisler:
 - <http://www.seas.gwu.edu/~bell>
 - <http://www.seas.gwu.edu/~cs210/cs210syllabus.htm>
 - Additional references include the following books:
 - Required textbook for Csci 210.
 - A comprehensive list of references can be found in the websites of Professors Bellaachia, Lancaster, and Kaisler.

- What format and time will the problems be?
 - A problem in the advanced software paradigms area will be expected to take one hour to solve in the average case.
 - A problem typically (but not always) consists of multiple parts, with each part addressing one (or more) specific topic(s). Thus, each problem is expected to cover a number of topics in the area. The goal of the question is usually to require that you demonstrate the ability to integrate multiple concepts relevant to developing large, complex software systems.
 - The questions may require the comparison of different programming languages.
 - The questions may require applying the principles from one or more software paradigms to design a solution to a specified problem
 - The questions may require comparison of some elements within a particular software paradigm.