COURSE DESCRIPTION

Dept., Number	CSci 550	Course Title	Program Semantics and Derivation
Semester hours	3	Course Coordinator	H. Conrad Cunningham, Professor

Current Catalog Description

A study of formal methods for the specification, derivation, and verification of computer programs. Predicate logic; notations for specification of programs; programming language semantics; calculational techniques for the derivation of programs

Textbooks

David Gries. *The Science of Programming.* Springer Verlag 1981.

H. C. Cunningham. *A Programmer's Introduction to Predicate Logic.* UMCIS-1994-02, revised 1996, reprinted January 2006. http://www.cs.olemiss.edu/~hcc/reports/prog_intro_logic.pdf

H. C. Cunningham. *Notes on Program Semantics and Derivation,* August 2006, http://www.cs.olemiss.edu/~hcc/csci550/notes//csci550_notes.pdf

References

Edward Cohen. *Programming in the 1990's.* Springer-Verlag, 1990. (This is a previous textbook that is out of print)

Class website: http://www.cs.olemiss.edu/~hcc/csci550

Course Outcomes

Upon successful completion of this course, the students:

1. know the concepts and terminology of predicate logic and guarded command semantics,
2. are able to apply logic to specify simple programs formally and prove them correct,
3. are able to apply basic program derivation techniques.

Relationship between Course Outcomes and Program Outcomes

This is a course taken primarily by beginning computer science graduate students; it is sometimes taken by undergraduate computer science students as an elective to enrich their programs. The course outcomes contribute to the program outcomes as follows: (1) to (a), (2) to (b), and (3) to (k).

Prerequisites by Topic

1. Basic discrete mathematics (Math 301)
2. Fundamental programming concepts and skills (CSci 112, 211)
3. Basic data structures and algorithms (CSci 112, 211)

Major Topics Covered in the Course

1. Introductory example (1 hour).
2. Predicate logic (8 hours).
3. Program specification techniques (3 hours).
4. Guarded Commands notation and its semantics (5 hours).
5. Program correctness verification (3 hours).
6. Program derivation techniques (1 hour)
7. Loopless programs (2 hours)
8. Loop heuristic "deleting a conjunct" (3 hours)
9. Loop heuristic "replace a constant by a variable" (9 hours)
10. Tail recursion (1 hour)
11. Exams (3 hours)

Assessment Plan for the Course

This is an elective course offered infrequently (e.g., only once in the past six years) and primarily to computer science graduate students. An offering typically has 3 examinations and 6 homework assignments. Outcome 1 is assessed by several exam questions and 2 homework assignments, outcome 2 by several exam questions and 2 homework assignments, and outcome 3 by several exam questions and 2 homework assignments. The instructor evaluates the student performance informally and makes changes to the course content, organization, and pedagogy as appropriate for subsequent offerings of the course.

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

Estimate Curriculum Category Content (Semester hours)

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