CS420: Program Derivation

Instructor: Om Damani

Course Objective

- Training in the art of 'Problem Solving' / 'Program Calculation'/ 'Calculating Implementation from Specification'
- In other words: 'Art of Writing Correct Programs'
- Specifically
  - Backward Reasoning:

    `Would you tell me, please, which way I ought to go from here?'

    That depends a good deal on where you want to get to,' said the Cat.
  - *Importance of Writing (using paper and pencil)*
  - Formal Specification
  - Syntactic Manipulation

References

1. Anne Kaldewaij, Programming: The Derivation of Algorithms, Prentice Hall International, 1990.

2. Edsger W. Dijkstra' work: Online archive - http://www.cs.utexas.edu/users/EWD/

Communication

- Moodle site: http://moodle.iitb.ac.in
  - Look for CS420
  - Includes newsgroup, mailing lists etc.

Course Load

- It is a 6 credit course
- I expect 9 hours of work every week: 3 hrs of lectures + 6 hrs of self-study and home-work
- Assignments every week for regular practice
  - Not all assignments will be graded
  - You will not know in advance which one will be graded :)

Evaluation

- End Sem: 40%
- Mid Sem: 20%
- Quizzes: 10%

- Assignments: 10%
  - No deadline waivers except for emergency

- Class Participation: 20%

- Relative Grading
  - But all can get AA or FF

Attendance Policy

- 90% attendance compulsory – will be taken after add/drop period ends
- Unavoidable reasons for non-attending must be communicated at the earliest

Office Hours: 24x7

- Instructor: TBD in consultation with students. In my office SIA 316
- I am available to you all the time. Call me at 932 300 3401 or x7948 or x8948 if needed

Feedback

- Two way feedback is critical to the success of the course:
  - Feedback to the students
    - HW, Quizzes, Exam
  - Feedback from the students
    - During the class – ask if something is not clear
      - Will help pace the lecture
      - Right after the class if a lecture was not clear at all
        - Will help decide if some concepts need to be rediscussed in the next lecture
    - While going through the lecture notes – make a list of questions that come to your mind, note down inaccuracies, unclear parts in the slide.
    - In person
    - Use email for subjective feedback, always ask technical questions in person or on newsgroup
  - Teaching evaluation forms

Academic Integrity

- Dishonesty implies straight FF
- Constructive discussion allowed, but if you copy from someone, you must acknowledge them explicitly
- You will not be cheating others but only yourself of your future

Syllabus

// Background: The Science

1. Problem Specification using Predicate Calculus

2. The Shape of Programs: assignment, sequential composition, conditional, and loop in The Guarded Command Language:

3. The Weakest Precondition

// **Techniques: The Heuristics**

4. Developing Loopless Programs: Calculating expressions in assignments

5. Developing Loops: Deciding on a loop invariant, taking conjuncts as invariants, replacing constants by variables, strengthening invariants, tail invariants, recursion.

6. Designing properly terminating constructs.

7. Shape of Mathematical Arguments: Exploit symmetry, avoid avoidable case-analysis.

// **Problems Covered**

8. Problem Domains: Searching, Segment problems, Array Manipulation, Sorting, Graph problems, other non-programming mathematical problems.

Pre-Requisites

- Background in Propositional Logic and Predicate Calculus
- Are the following statements **true**
  - p => ~p => ~p => p
  - p => q => p
- Given: (*this example is taken from Prof. Cunningham's notes*)
    - taken.s.c : Student s has completed class c.
    - passed.s.c : Student s received a passing grade in class c.
    - grad.s : Student s is a graduate student.
    - cunningham.c : Class c is taught by Professor Cunningham
    - theory.c : Class c is a theory class.
  - Write a logical formula representing
    - Every graduate student took one of Professor Cunningham’s classes and passed one theory class

If you find the above difficult, then this course may not be for you