CPSC/ECE 3520:
Programming Systems
Summer Session II 2017
Extended Syllabus

Dr. Robert J. Schalkoff
Professor
Electrical and Computer Engineering
334 Fluor Daniel Engineering Innovation Building
Clemson University
Clemson, SC 29634-0915
rjschal@g.clemson.edu
rev. 5-23-2017 *

*Note that this syllabus will be accompanied by a video lecture once the session begins.
# Contents

1. **Course Overview / Goals and Objectives**  
   1.1 Course Description  
   1.2 Course Pragmatics  
   1.3 Course Progress  
   1.4 Attendance  
   1.5 Prerequisites  

2. **Course Outline by Topics (in approximate chronological order)**  

3. **Course Schedule**  
   3.1 Session Summary Calendar 'at a Glance'  
   3.2 Master Calendar of CPSC/ECE 3520 Lectures (Slides and Videos)  
   3.3 Critical Summer Session II Dates for CPSC/ECE3520  
   3.4 Daily Reminder Via email  

4. **Software Development**  
   4.1 SDEs  
   4.2 SDE Implementation Tools  

5. **Course Grading Procedures**  
   5.1 Quizzes and SDEs  
   5.2 Students with Disabilities: Requesting Accommodations  
   5.3 Proctors  

6. **Course Structure**  
   6.1 Modules  
   6.2 Course Content by Module  
   6.3 The Master Calendar a.k.a. Slide-Video Catalog (Again)
7 Additional Remarks and Frequently Asked Questions

7.1 Incorporation .................................................. 13
7.2 Interaction Notice ............................................. 13
7.3 Title IX Statement for Undergraduate Syllabi ............ 13
7.4 Teaching Philosophy .......................................... 14
7.5 General FAQs ................................................... 14
1 Course Overview / Goals and Objectives

1.1 Course Description

CPSC/ECE 3520 is a more advanced course in programming languages and systems for computer engineering/science majors. The objective of this course is to enable a more complete understanding of software development topics and related supporting tools, including philosophy, methodology, formal syntax and semantics, and examples of programming paradigms, languages and development approaches.

1.2 Course Pragmatics

Instructor: Dr. R.J. Schalkoff

If you want to see a recent picture of me with some background information, please visit:

http://www.clemson.edu/ces/departments/ece/faculty_staff/faculty/rschalkoff.html

Office: EIB 334, 864-656-5913, rjschal@g.clemson.edu

Web Page: We will be using the Canvas system extensively. You will find this syllabus, as well as related materials (course slides and lecture videos) on the CPSC/ECE 3520 Canvas page. This repository will evolve over the session. The all-important course ’Master Calendar’ may be found as there. Be sure you have access to this course on Canvas, since that is how you will receive lectures (slides and videos), software assignments and other relevant information. You will also upload SDE assignments on Canvas.

At this time, all accompanying course videos are in Flash Video, i.e., flv file format. As noted later in this document, you should be using a linux OS for this course. To this end, I recommend the linux version of vlc as the preferred video viewer.
Canvas Pragmatics: If you are enrolled in this course, you should be able to see this course as an option when you login to Canvas. The Canvas pages are at:

http://www.clemson.edu/canvas/

For those new to canvas, you might want to look at the Canvas Resources, especially the 'Canvas Student Guide' in the Resources Section:

http://www.clemson.edu/canvas/resources.html

or the Full Student Guide at:

https://community.canvaslms.com/docs/DOC-4121

Text: Schalkoff, Programming Languages and Methodologies, Jones and Bartlett Publishers, 2007. Required. Note: The least expensive and easiest way to get this is via Campus Copy Shop in Clemson:

Campus Copy Shop
384-1 College Ave.
Clemson, SC 29631
Phone: 864-654-3863
Office Hours: M,T,W,Th,F 11:00AM-12:00PM (by phone), unless notified otherwise¹. Others by appointment. You can contact me by email (rjschal@g.clemson.edu) anytime.

1.3 Course Progress

Although I will try to help you stay in sync with this course, a very important point is:

You and I are ’in this together’ for the session. However, your progress is up to you.

1.4 Attendance

ECE/CPSC3520 is an on-line course. There will be no meetings of this course in a traditional classroom. However, you should be checking your email and Canvas daily. It is of utmost importance to be following the ’Master Calendar of CPSC/ECE 3520 Lectures’.

1.5 Prerequisites

CPSC 3520: CPSC 212/215
ECE 3520: ECE 223

2 Course Outline by Topics (in approximate chronological order)

1. Programming: the Concept, History and Trends
   
   (a) Communication with a machine: what does ’language’ have to do with it?
   
   (b) How do you Convey Aspects of a Programming Language?

¹Due to a preexisting commitment, this will be modified to be email-only from 6-28 through 6-30. Note July 4 is a holiday.
(c) Taxonomy and History of Programming Languages and Approaches

2. Specifying and Enforcing Syntax
   (a) Grammars
   (b) BNF Notation and Alternatives
   (c) Parsing
   (d) "mini-c"

3. Prolog, Prolog Applications and Parsing in Prolog

4. Functional Programming Approaches
   (a) The Lambda Calculus
   (b) ocaml

5. Specifying Semantics
   (a) Why?
   (b) Relationship to "Correctness"
   (c) Translational Semantics
   (d) Operational Semantics
   (e) Denotational Semantics
   (f) Axiomatic Semantics

6. Copyright and Software Licenses

3 Course Schedule

As noted, numerous attempts and tools will be used to help keep you ’on schedule’ throughout the session. These are explained below.
3.1 Session Summary Calendar 'at a Glance'

3520 SSII 2017 ('at a glance')

June 2017

Su Mo Tu We Th Fr Sa
28 29 30 31 Start June 28 (lecture 1)
Proctor forms are due June 30

July 2017

Su Mo Tu We Th Fr Sa
1
2 3 4 5 6 7 8 July 4 is a holiday
9 10 11 12 13 14 15 Quiz #1 is July 12
16 17 18 19 20 21 22 SDE1 is due July 18
23 24 25 26 27 28 29 Quiz #2 is July 26
30 31 SDE2 is due July 31

August 2017

Su Mo Tu We Th Fr Sa
1 2 3 4 Quiz #3 is Aug 4 (final)

3.2 Master Calendar of CPSC/ECE 3520 Lectures (Slides and Videos)

The course Canvas page contains a page with a more detailed 'Master Calendar of CPSC/ECE 3520 Lectures' PDF indicating posted day-by-day lectures, deliverables and significant course activities. It is of utmost importance to be either in sync or slightly ahead of lecture pace indicated by this Course Master Calendar.

The Course Master Calendar file is:
3520-slide-video-schedule-SSII2017.pdf

You should download this file from Canvas and post it in a prominent place.

3.3 Critical Summer Session II Dates for CPSC/ECE3520

These are probably the most important dates to put on your 2017
calendar.

- Wednesday 28 June: Course Starts
- Friday 30 June: Proctor Approval Form Due (See below and Canvas Content item)
- Tuesday July 4: Holiday
- Friday July 7: SDE #1 Assigned
- Wednesday July 12: Quiz #1
- Tuesday July 18: SDE #1 Due
- Friday July 21: SDE #2 Assigned
- Wednesday July 26: Quiz #2
- Monday July 31: SDE #2 Due
- Friday August 4: Quiz #3; Course Ends

3.4 Daily Reminder Via email

You will get email from me every morning (on a 'course' day) with a header something like:

rjschal-3520SSII2017: <date>: Goals and Notes

indicating previous, current and future CPSC/ECE 3520 issues. These emails are aligned with the course Master Calendar. This is to help keep you up-to-date with respect to the CPSC/ECE 3520 course schedule. In the past, students have indicated these emails are helpful, so I will continue the effort. They are also cataloged online on Canvas as 'Announcements'.

4 Software Development

4.1 SDEs

You will undertake 2 Software Design Exercises (SDEs) in CPSC/ECE 3520 as part of your graded submissions. They will be graded on the basis of
functionality, completeness and correctness and must be submitted by the due date. (Please read the previous sentence again.) For some students, the SDEs are the most challenging part of the course. For some students, the SDEs are the most rewarding part of the course. Your motivation and time management will determine which one it is for you.

There is also an early, non-graded, but required ‘SDE0’, which is just a test of your ability to upload an archive to Canvas. It will be posted on Canvas and should only take a few minutes to complete.

4.2 SDE Implementation Tools

There will be 2 graded Software Development Exercises (SDEs). To this end, two things are important:

1. You should be using a linux-based development environment. Any modern linux distribution will suffice. You may even run linux in a virtual environment.

2. A key aspect of software development is time management and meeting deadlines. The due date for each SDE is not a recommendation, it is a firm deadline.

Languages/software tools we will use for the 2 SDEs are:

1. SWI-Prolog (http://www.swi-prolog.org/)

2. ocaml (http://caml.inria.fr/ and/or https://ocaml.org/)

You are not expected to be familiar with these paradigms/languages. Both are open source and free. It would not hurt to have these software packages installed prior to the course start, in order to allow more time during the session for the SDE completion. I’ll point out in the lectures (and the book) where to get them and how to install them.

To help guide you towards a successful SDE, developed in a bottom-up fashion, I will specify deliverable predicates (Prolog) or functions (ocaml). They range from very simple to the top-level, and you should develop them in that order.
5 Course Grading Procedures

5.1 Quizzes and SDEs

There will be 3 quizzes and 2 SDEs. They will determine 100% of your final grade. Thus there are 5 scores recorded. Quizzes are equally weighted. No re-tests are given, and no additional work will be assigned. I will use all 5 scores recorded in computing your final grade\(^2\).

5.2 Students with Disabilities: Requesting Accommodations

Students with disabilities requesting accommodations should make an appointment with the Director of Disability Services (656-6848), to discuss specific needs immediately.

5.3 Proctors

- If you will be on or near campus (within 25 miles), you will take the quizzes in a classroom.

- If you will be off-campus, you will need to find a proctor who is willing to supervise your taking of the quizzes. We will send the quiz to the proctor the night before the quiz date via email as a PDF attachment. The proctor will be asked to provide you with a quiet place to take the quiz, enforce the rules of the quiz, scan your solution and send it back to the course grader (and 'proctor manager') via email as a PDF attachment\(^3\).

- A proctor must be a person of trust who agrees to supervise your taking of quizzes. Basically, the proctor must be a person you trust to be responsible and I have reason to trust. You must trust them to be available to receive, administer and return the quiz to me in a timely manner.\(^2\)

\(^2\)In other words, no scores are dropped.

\(^3\)Please note that, due to course enrollment, we will not be accepting FAXed quizzes. Keep this in mind when nominating your proctor.
manner. I need to be reasonably sure they will enforce the constraints on taking the quiz.

- Detailed information regarding qualifications and duties of a proctor are described in the ECE proctor information and approval form, specifically the section 'Online Summer Program Proctor Information' which is posted as a page on Canvas. You are to recommend a proctor using the Proctor Approval/Agreement Form. Fill out this form in concert with your proctor, scan it, and submit it via Canvas no later than Friday, June 30.

6 Course Structure

6.1 Modules

Shortly before the class begins, I will begin posting course Modules and Pages on Canvas. A module consists of some combination of related book reading, slide sets, video sets, auxiliary references, web references, code examples, and other material.

6.2 Course Content by Module

1. Introduction

2. Grammars/syntax/derivation/parse trees/parsing(CYK)

3. Declarative programming in Prolog

4. Functional programming in ocaml

5. Programming Language (Formal) Semantics and Software Licenses
6.3 The Master Calendar a.k.a. Slide-Video Catalog (Again)

To coordinate module lecture topics and slides with associated lecture videos and recommended viewing dates, a file: 3520-slide-video-schedule-SSII2017.pdf is available on Canvas. This document is your course 'lifeline', and provides a reference on where you should be lecture-wise at any point in the session.

7 Additional Remarks and Frequently Asked Questions

7.1 Incorporation

The Clemson Announcements contains additional information and guidelines on a number of important and related topics, including special needs and academic integrity. These guidelines are incorporated into CPSC/ECE 3520 by reference.

7.2 Interaction Notice

In this online course, you will interact with the content and instructor on at least a weekly basis through course assignments, asynchronous discussions, and/or synchronous sessions as indicated in the syllabus.

7.3 Title IX Statement for Undergraduate Syllabi

Clemson University is committed to a policy of equal opportunity for all persons and does not discriminate on the basis of race, color, religion, sex, sexual orientation, gender, pregnancy, national origin, age, disability, veterans status, genetic information or protected activity (e.g., opposition to prohibited discrimination or participation in any complaint process, etc.) in employment, educational programs and activities, admissions and finan-
cial aid. This includes a prohibition against sexual harassment and sexual violence as mandated by Title IX of the Education Amendments of 1972. This policy is located at http://www.clemson.edu/campus-life/campus-services/access/title-ix/. Mr. Jerry Knighton is the Clemson University Title IX Coordinator. He also is the Director of Access and Equity. His office is located at 111 Holtzendorff Hall, 864.656.3181 (voice) or 864.565.0899 (TDD).

7.4 Teaching Philosophy

My teaching philosophy may be summarized in these remarks:

- The class (you) and I are 'in this together' for the session. However, your progress is up to you.
- Sometimes we can have fun with the material; sometimes work is required. Not everything can be made simple or fun.
- Academic integrity is a serious issue.
- There are standards for academic achievement, and they should be employed. Excessive whining has never been shown to facilitate the installation, creation or debugging of software.
- All our interactions should be conducted within a framework of mutual respect.
- The time to put effort into 3520 is before the quizzes, SDEs and before the course is over.

7.5 General FAQs

The course Canvas page contains several FAQs, including a more comprehensive version of the General FAQ below. Each SDE will have an associated FAQ, summarizing previously asked questions. Be sure to check the FAQs frequently since they are updated throughout the session.

- What Should I Expect?
• How Hard is The Class?

• What is Required?
  - This class requires active participation.
  - This class requires self-motivation and (sometimes) perseverance.
  - Remark:
    
    I am not here to simply 'help you'.

    but ...

    I am here to help you learn how to help yourself.
  - This class requires following instructions and meeting (SDE) deadlines.
  - This class requires installing software and reading manuals. ('Read the manual').
  - This class requires designing, implementing, testing and submitting software that works correctly. (A surprise to some?).

• Any Helpful Hints for Success?
  - Stay up to date (or even slightly ahead) with the lectures.
  - Read the book.
  - Pay attention to the examples (book and lectures). Try the examples. Modify the examples.
  - Read the SDE assignment documents promptly, carefully and repeatedly.
  - Plan your SDE effort. ('It is never to early to get started'). Beginning the night before the SDE is due is usually a contraindication to success.
  - Do not approach software development with an imperative (c or java or matlab or perl ...) mindset.