CAAM 519 Fall 2018 Syllabus

Loïc Cappanera

1 Course Information

Class Times: 11:00am to 11:50am on Monday, Wednesday, and Friday.
Course Schedule: 20th August to 30th November 2018.
Location: Duncall Hall 1042.
Instructor: Loïc Cappanera
Contact: loic.cappanera AT rice.edu
Office Hours: 1:00pm to 2:00pm Tuesday in DCH 2001.
Teacher Assistant: TBA
Contact: TBA
Office Hours: TBA.


2 Course Description

The goal of this course is to make the student familiar with a set of basic tools for solving problem in scientific computing. It includes high-level programming languages (C and C++), program development and maintenance tools (make, debuggers, version control systems) and typesetting utilities (LaTeX). We will work exclusively in a Unix-like environment (Linux, Mac OS-X). The course itself can be splitted in three sections that are described in the following.

2.1 General programming tool

The first few weeks of the course focus on introducing general tools commonly used when programming. This course section can be divided into three areas:

- Working in a Linux environment. Introduction to Linux commands, access to remote machine/cluster with ssh, use of various text editors and virtual machine with VirtualBox.
- Using version control system. Introduction to GIT and SVN softwares.
• Writing scientific document and oral presentation with \LaTeX.

2.2 The C programming language

The second part of the course covers the following aspects of C:
• Basic types and operations.
• Control structures.
• IO (input/output) and formatting.
• Arrays and pointers.
• Memory allocation for arrays.
• Structures and typedef.

As developing programs often requires the use of external libraries and always gives rise to bug, we also introduce general tools not specific to C language:
• make (generation of executable program including libraries).
• gbd (debugger).
• valgrind (tool to fix memory leak issue).

2.3 The C++ programming language

The last part of the course is an introduction to C++. This language is a superset of C that enhances many of C language’s features (better memory management, add of exception to handle bug, etc.) However the main novelty of C++ is the possibility to do object-orientated programming. For instance, it allows to write solver that can work with a various range of objects (such as various finite element meshes, finite element spaces, multidimensional arrays, etc.). Note that C++ is not the only language that allows object-orientated programming (see Java, Fortran 2003, Python and others). The agenda for this part of the course covers the following aspects of C++:
• Classes.
• Memory management, constructors and destructors, access control.
• Inheritance and virtual functions.
• Templates and the standard library.

3 Objectives and Outcome

3.1 Objectives

CAAM 519 students learn to design and implement concepts, algorithms and methods derived from applied mathematics using the C and C++ programming languages.
3.2 Course Outcomes

After completing this course, students should be able to:

- Navigate comfortably within the Linux operating system.
- Manage a computing project with the Subversion source management system.
- Use \LaTeX{} to write paper or oral presentation.
- Manage to use visualization tools such as Gnuplot and Paraview.
- Create C/C++ computer programs for mathematical applications.
- Use the GNU make system (i.e. create and use Makefiles).
- Debug C/C++ programs using the GNU debugger gdb.
- Find memory errors in C/C++ programs using Valgrind.
- Validate and verify C/C++ programs.
- Use mathematical libraries (BLAS, LAPACK, FFTW, UMFPACK).

4 Materials

4.1 Texts

There are no required texts for the course, and all material will be provided online as notes and slides by the instructor. The recommended reference texts are *The C Programming Language* by Brian W. Kernighan and Dennis M. Ritchie, *The C++ Programming Language* by Bjarne Stroustrup and *The \LaTeX{} Companion* by Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley.

4.2 Computer

It will not be possible to complete this course without access to a computer for coding and running assignments. If a student does not currently have access, they must contact the instructor and arrange an alternative by the first week of class.

5 Disability Support Services

Any student with a disability requiring accommodation in this course is encouraged to contact the instructor during the first week of class, and also to contact Disability Support Services (Allen Center, Room 111 / adarice@rice.edu) to determine the accommodations you need.