Syllabus for:
Introduction to Scientific Programming
SDS 322/392, 57330/57605

Victor Eijkhout and Charlie Dey

Spring 2017

Time and place	FAC 101B	15:30–17:00 TTh
Listed instructor:	Victor Eijkhout	eijkhout@tacc.utexas.edu, ROC 1.425, 471-5809
Co-instructor:	Charlie Dey	charlie@tacc.utexas.edu, ROC 1.114, 232-7591
Office Hours:		TBD
TA:	Gavin Liu	zhuode93@cs.utexas.edu, hours TBD

Contents

1 Rationale

Computers were invented over 60 years ago to solve mathematical equations, especially in science and engineering. Over the last 20 years, computers have become ubiq-

uitous in our lives: increasing productivity, enhancing communications and connectivity, ensuring safety, and providing entertainment. The pervasiveness of computers in business and consumer environments – and the resulting revenues – has caused a shift in the computer languages and skills taught in computer science departments today. Introductory programming classes are now commonly taught in Java, and focus on skills needed for industry careers in web development, at the expense of offering scientific computing classes formerly taught for the benefit of technical computing – science and engineering departments at universities as well as the R&D departments in numerous industries. This has occurred even though computers have become increasingly fundamental to the conduct of science and engineering, two fields vital for increasing economic productivity, ensuring national security, and addressing many important societal problems.

There are recent signs of a reversal of this trend: new degree programs in computational science, new emphasis in federal spending on computational technologies and R&D, and even a return of some scientific computing classes in some computer science departments, etc. However, the need for well-trained computational scientists and engineers is still urgent. Fortran, C and C++ remain the most powerful general purpose programming languages for developing scientific software: they offer the best features and flexibility for designing robust, high-performance applications. These languages are constantly evolving and growing, new trends and concepts for software development in both languages frequently arise and are rapidly disseminated. The SDS 322/392 course provides a unique opportunity to learn these languages

2 Course Aims and Objectives

2.1 Course organization

In this course you will learn the process of developing scientific applications, by successively learning to program in C++, Fortran, and to a small extent Python.

There will be tutorials, especially early in the course, on the use of Unix, and programmer tools.

Programming language learning will be project-driven, given you quick exposure to the basics of

1. data types
2. expressions
3. control structures
4. scoped constructs such as functions
5. classes and objects
6. arrays

We will briefly touch on advanced concepts such as

- User-defined data structures

- Modular programming
- Dynamic memory
- Templates
- Input/output
- Libraries

By the end of this course, the students will:

- have understanding of the commonly used scientific programming languages C++ and Fortran;
- have their programming and problem-solving skills
- understand the process of debugging the code;
- have familiarity with the Linux Operating System.

3 Instructors Biographical Information

Victor Eijkhout Victor is a research scientist in the High Performance Computing group at the Texas Advanced Computing Center (TACC). His degree is in numerical analysis, and he has longtime experience in programming scientific codes and libraries, especially on parallel computers. His research interests include parallel computing, machine learning, processor performance. He has written several widely used textbooks, as well as many scientific papers.

Charlie Dey Charlie is a software engineer, educator and researcher with the User Services group at the Texas Advanced Computing Center (TACC). He is involved in many aspects of TACCcurrently involved on an NSF project known as DesignSafe, a cloud resource supporting the entire scientific workflow for the natural hazards community. . Charlie is also involved in breast cancer research, solving a vast system of equations by applying a numerical method he developed to solve highly stiff nonlinear partial differential equations without resorting to smaller step sizes. Charlie has been involved in many facets of computer science and its applications - healthcare, banking, manufacturing, education, gaming, and HPC; his current areas of interest are numerical analysis, mathematical modeling, scientific visualization, and education and outreach.

4 Format and Procedures

We encourage a lively participation during the lectures and expect that you participate by asking and answering questions. Active participation makes for a better and more interesting class for you and for us, and allows us to assess your progress and to adjust the class material and/or teaching progress accordingly.

Student progress in this class will be evaluated through quizzes, homework, a final project, and a final exam. We expect timely notification if you cannot take a quiz/exam as scheduled, or if you are unable to meet the deadline for the homework or the final project.

5 Other course information

5.1 Prerequisites

Students will be given access to a unix-based machine with C/C++ and Fortran compilers. Students who do not know Unix need to do a tutorial as soon as possible.

Familiarity with basic mathematics is assumed.

5.2 Suggested Course Readings/Materials

- Modern Fortran Explained, by Michael Metcalf, John Reid, Malcolm Cohen (*Older versions do not explain all language features*)
- C++ Primer Plus (sixth edition), Stephen Prata (*Older versions do not explain all language features*)

5.3 Computing Resources

The classroom resources will be based on a virtual machine hosted on TACCs VM farm known as Rodeo. On day 1, Students will sign up for a TACC user account that will be used to login to the class VM. The virtual machine will be loaded will the all the necessary software for this course and students will be required to do all their labs and assignments on this machice.

5.4 Piazza for discussions

Students can post discussion or questions and suggest answers on Piazza: go to `https://piazza.com` and enroll with your utexas email. Instructors will make an effort to check the forum at least once a day. Ask your questions here before mailing the instructor or TA!

6 Grading Procedures

- There will be homework exercises and exams, both counting for 50 points.
- There will be two midterm exams for 30 points each, and a final exam in the form of a programming project for 40 points.

Participation in class and on piazza may cause your grade to be rounded up.

7 Formal and informal policies

Class attendance and participation policy

We expect students to attend and participate in class in accordance with the UT Honor Code (see below). Students are encouraged to ask questions, especially relating to material used in their projects.

This class will have lecture and lab sessions. During the lectures there are to be no laptops open, as these are distracting to the student, as well as others who can see the screen. Tablets with non-raised screens are allowed.

7.1 Class attendance and participation policy

- We expect students to attend and participate in class in accordance with the UT Honor Code (see below)
- Students are encouraged to ask questions.

7.2 Religious Holy Days

By UT Austin policy, you must notify us of your pending absence at least fourteen days prior to the date of observance of a religious holy day. If you must miss a class, an examination, a work assignment, or a project in order to observe a religious holy day, we will give you an opportunity to complete the missed work within a reasonable time after the absence.

7.3 Academic Integrity

University of Texas Honor Code
The core values of The University of Texas at Austin are learning, discovery, freedom, leadership, individual opportunity, and responsibility. Each member of the university is expected to uphold these values through integrity, honesty, trust, fairness, and respect toward peers and community.

Each student in this course is expected to abide by the University of Texas Honor Code. [See the UT Honor Code above.] Any work submitted by a student in this course for academic credit will be the student's own work. Collaborations will be allowed for the course project.

You are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give "consulting" help to or receive "consulting" help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e-mail, an e-mail attachment file, a diskette, or a hard copy.

Should copying occur, both the student who copied work from another student and the student who gave material to be copied will both automatically receive a zero for the assignment. Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result

in failure of the exam, and may lead to failure of the course and University disciplinary action.

7.4 Other University Notices and Policies

Students with Special Concerns
Students with special concerns - be they athletes who might miss class meetings, students with religious observances that interfere with class meetings, or students with disabilities who need special accommodation - are all supposed to notify us about these special needs by the 12th class day which is 13th September 2013.

Use of E-mail for Official Correspondence to Students
All students should become familiar with the University's official e-mail student notification policy. It is the student's responsibility to keep the University informed as to changes in his or her e-mail address. Students are expected to check e-mail on a frequent and regular basis in order to stay current with University-related communications, recognizing that certain communications may be time-critical. It is recommended that e-mail be checked daily, but at a minimum, twice per week. The complete text of this policy and instructions for updating your e-mail address are available at http://www.utexas.edu/its/help/utmail/1564 .

Documented Disability Statement
Any student with a documented disability who requires academic accommodations should contact Services for Students with Disabilities (SSD) at (512) 471-6259 (voice) or 1-866-329-3986 (video phone). Faculty is not required to provide accommodations without an official accommodation letter from SSD. Please notify us as quickly as possible if the material being presented in class is not accessible (e.g., instructional videos need captioning, course packets are not readable for proper alternative text conversion, etc.).

Please notify us as early in the semester as possible if disability-related accommodations for field trips are required. Advanced notice will permit the arrangement of accommodations on the given day (e.g., transportation, site accessibility, etc.).

Contact Services for Students with Disabilities at 471-6259 (voice) or 1-866-329-3986 (video phone) or reference SSDs website for more disability-related information: http://www.utexas.edu/diversity/ddce/ssd/for_cstudents.php

Behavior Concerns Advice Line (BCAL)
If you are worried about someone who is acting differently, you may use the Behavior Concerns Advice Line to discuss by phone your concerns about another individuals behavior. This service is provided through a partnership among the Office of the Dean of Students, the Counseling and Mental Health Center (CMHC), the Employee Assistance Program (EAP), and The University of Texas Police Department (UTPD). Call 512-232-5050 or visit http://www.utexas.edu/safety/bcal.

Drop Policy

The State of Texas has enacted a law that limits the number of course drops for academic reasons to six (6). As stated in Senate Bill 1231:

> Beginning with the fall 2007 academic term, an institution of higher education may not permit an undergraduate student a total of more than six dropped courses, including any course a transfer student has dropped at another institution of higher education, unless the student shows good cause for dropping more than that number.

Emergency Evacuation Policy
Occupants of buildings on the UT Austin campus are required to evacuate and assemble outside when a fire alarm is activated or an announcement is made. Please be aware of the following policies regarding evacuation: * Familiarize yourself with all exit doors of the classroom and the building. Remember that the nearest exit door may not be the one you used when you entered the building. * If you require assistance to evacuate, inform us in writing during the first week of class. * In the event of an evacuation, follow the instructions of class instructors. Do not re-enter a building unless youre given instructions by the Austin Fire Department, the UT Austin Police Department, or the Fire Prevention Services office.