Syllabus (Fall 2012)

1 Official Yale course listing

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Instructor</th>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPSC 427</td>
<td>Object-Oriented Programming</td>
<td>Michael Fischer</td>
<td>TTh 2.30–3.45</td>
<td>Fall 2012</td>
</tr>
<tr>
<td>CPSC 527</td>
<td>Object-Oriented Programming</td>
<td>Michael Fischer</td>
<td>TTh 2.30–3.45</td>
<td>Final exam scheduled (Group 27)</td>
</tr>
</tbody>
</table>

Object-oriented programming as a means to efficient, reliable, modular, reusable code. Use of classes, derivation, templates, name-hiding, exceptions, polymorphic functions, and other features of C++.

2 Course Description

2.1 Nature and Purpose of the Course

Programming can be viewed in two ways: It is the activity of instructing the computer in the solution of a particular problem, or it is the activity of constructing a piece of software with desired properties. The former view focuses on the algorithmic process for producing the answer. It is what is typically learned in beginning programming courses. The latter view focuses on all aspects of the software – its size, speed, complexity, stability, correctness, reliability, clarity, maintainability, security, ease of repurposing, and ease of deployment – as well as on solving the problem.

This course is concerned with software architecture, design, verification, and construction. It exposes students to modern object-oriented programming techniques that have proved successful in the development of large complex software systems by multiple programmers. It teaches object-oriented design in C++ and explores techniques for building modular, efficient and robust systems.

2.2 Main Topics to be Covered


3 Course materials

Required textbook:

Recommended reference:


Other materials will be posted on the course web site from time to time during the term.

Website: I maintain a course website at [http://zoo.cs.yale.edu/classes/cs427/2012a/index.html](http://zoo.cs.yale.edu/classes/cs427/2012a/index.html). You should bookmark it in your browser and visit it often. It will grow as the term progresses and will contain announcements, handouts, homework assignments, programming hints, and links to documents in the course directory and elsewhere on the web. Access may be restricted to machines on the Yale network. If so, for off-campus use, you will need to follow the instructions for accessing Yale resources from off campus.

4 Course Mechanics

**Prerequisite:** The prerequisite for this course is CPSC 223 (Data Structures) or equivalent. The ability to write significant programs in the C programming language is required. The course also assumes a familiarity with basic computer science concepts such as are covered in CPSC 201 and CPSC 202.

**Requirements:** Course requirements include weekly programming assignments and/or written problem sets (~50%), a midterm exam (~20%), and a final exam (~30%). The approximate weights of each in determining the course grade are subject to change depending on the number and difficulty of the assignments actually given. Graduate students taking the course will be expected to perform at a higher level than undergraduates and may be required to do additional work and/or presentations.

**Assignments and Other Announcements:** Written problem sets and programming assignments will be posted on the handouts page of the course web site from time to time during the course. Other course announcements will be posted on the course home page. It is your responsibility to check these pages frequently.

**Help with Technical Material:** The graduate teaching assistants and undergraduate course graders will be holding scheduled office hours during the term, which will be posted on the course home page. You are encouraged to meet with them with questions about the lectures, textbook, problem sets, and OO-C++ programming generally. You may also send questions to the instructional staff by email. For fastest response, please use the email address cs427help@cs.yale.edu. Your question will go to the entire instructional staff, and whoever is available at the time can decide to answer it. The response will also go to the entire staff so the others will know that it has been taken care of.

**Other Questions:** All questions about assessment and grading should be sent to the teaching assistants at cs427ta@cs.yale.edu (This email also goes to me.) If the TA’s are unable to resolve your questions to your satisfaction, or if you wish to talk to me privately about any matter, then you
are always welcome to contact me, either by email to fischer-michael@cs.yale.edu or in person. Email is also the preferred way to arrange an appointment with me.

5 Policies

Late Policy: Assignments will be due at 11:59 pm on the night of the stated due date. Late work will generally be subject to a penalty of 5% per day late unless accompanied by a Dean’s excuse. In counting the number of days late, there will be a two hour grace period between 11:59 pm at night and 1:59 am the following morning during which an additional day of late penalty will not be assessed.

Work more than 4 days late will not be accepted, but alternative means for making up missed work may be arranged on an individual basis with a Dean’s excuse.

Please contact the instructor as soon as you find out that you are unable to submit work on time or to attend a scheduled exam so that suitable makeup arrangements can be made.

Policy on Working Together: This course follows the Yale College policy on Cheating, Plagiarism, and Documentation with which you should familiarize yourself. Briefly, if you use someone else’s work, you must acknowledge it. If it’s a piece of code, place the acknowledgement in your source file and explain clearly what parts are not your own. Similarly, if it’s in a paper, the acknowledgement belongs in the paper itself. All work not so acknowledged must be your own.

You may of course discuss the lectures and readings with your classmates in order to improve your understanding of the subject matter. Helping each other learn to use the tools in the Zoo is also okay. However, the design and implementation of all programs and all written work must be your own except where other sources are explicitly noted.

You are always free (and encouraged) to come in and ask the TA or instructor for help about anything concerning the course. Please talk to the instructor if you have any questions about this policy.

Avoiding Plagiarism: You may neither copy from another student nor permit your own work to be copied, unless explicit permission is given for such collaborations. To avoid unintended involvement in plagiarism, your work should never be in the possession of another student. Do not ask someone else to deliver or pick up your work. Do not let another student “borrow” your code to compare with theirs. Keep your files protected so that others cannot read them and carefully guard your password. Do not leave printed work in public areas such as the Zoo or in accessible wastebaskets. If you think your password may have been compromised, you should change it immediately.

Policy on Computer Problems: The Yale College policy on Use of Computers and Postponement of Work in the Yale College Programs of Study applies to this course. It is reproduced below.

“Problems that may arise from the use of computers, software, and printers normally are not considered legitimate reasons for the postponement of work. A student who uses computers is responsible for operating them properly and completing work on time. (It is expected that a student will exercise reasonable prudence to safeguard materials, including saving data on removable disks at frequent intervals and making duplicate copies of work files.) Any computer work should be completed well in advance of the deadline in order to avoid last-minute technical problems as well as delays caused by heavy demand on shared computer resources in Yale College.”
Particularly relevant for this course are the cautions against leaving a programming assignment to the last minute when machines might be busy, printers broken, and so forth, and about safeguarding your data.

6 Computing Facilities

The Zoo: This course will use the Computer Science Department’s educational computing facility, affectionately known as the Zoo. This facility contains modern workstations SuSE Linux. You will need to use these machines to prepare and submit coursework. Look at

http://zoo.cs.yale.edu/help/

for information on getting started if you are new to the Zoo.

These days, most of you have your own laptops and may be wondering why you should be bothered with using a new computer system. The answer is because code development software is still not completely compatible across multiple platforms. If it works on your Mac or Windows PC but fails when the graders run it on the Zoo, you will lose points. If you ask for help with compiler errors on your personal machine, we won’t be in a position to answer your questions. In short, develop your code on the Zoo! Regardless of where the code is developed, your assignments must be submitted from your Zoo course account, and they will be graded according to how well they work on the Zoo.

The Zoo machines support remote access via the SSH and VNC protocols. These enable you to do your work remotely when it is inconvenient to go in person to the Zoo. Instructions on how to configure your machine for remote access will be posted to the course web site.

Your course account: You must request a course account for this course even if you already have a Zoo account. You will be unable to submit your assignments without it. To obtain your account, go to

http://zoo.cs.yale.edu/help/accessing-zoo.shtml

and follow the instructions there. Do not wait. Do it now. I will be unsympathetic for late submissions due your not having followed this instruction.

Course directory: The shared course directory, /c/cs427, is located on the Zoo server. You can access it from your Zoo course account. It will contain any software needed for this course and miscellaneous documentation and files. It will also contain software to allow you to submit assignments electronically. Public files there can be accessed via the web as well as from a Zoo node. Your class account home directories will also be located there.