Syllabus, CSC1254/1, Spring 2015

Introduction to Programming with C++ (II), Hartmut Kaiser
http://www.cct.lsu.edu/~hkaiser/spring_2015/csc1254.html

COURSE SUMMARY

This course is for someone who has had some programming experience before but nevertheless is willing to work hard to learn. It helps you understand the principles and acquire the practical skills of programming using the C++ programming language. The course is ideal for undergraduate research positions or summer jobs requiring C++. It is not a class for experienced programmers in C++.

My aim is for you to gain sufficient knowledge and experience to perform simple useful programming tasks using the best up-to-date techniques. Don’t expect to spend less time than maybe 15 hours a week for 14 weeks if you work by yourself.

The new C++11 international standard has finally being accepted last year. This is good news as it allows me to base all the course material on this new and revised language specification. Don’t be afraid, though. This will make your job easier. The revised C++ language is much easier to teach and to learn as many of the darker corners have been straightened out, and the overall appearance of the language is more homogeneous.

The approach taken in this course is “depth first” in the sense that it quickly moves through a series of basic techniques, concepts, and language supports before broadening out for a more complete understanding. Roughly the first half of the course concentrates on examples of using the standard library abstractions. In the second half of the course we will talk about how you can define your own abstractions. The general focus of the course is on data structures and algorithms, which means we will analyze existing ones, and later on, build our own data structures and algorithms while using the existing ones as a starting point.

The first lectures (about 5 weeks) cover objects, types and values, computation, debugging, error handling, the development of a couple of examples. This part of the course is mainly a review of language-technical aspects including the design of functions and classes. The C++ standard library is highlighted; we will explore how to utilize these predefined facilities. At the same time these lectures go beyond a simple review by discussing concepts, more details, and more in depth material.

The rest of the course is dedicated to the analysis of existing data structures and algorithms, such like vectors, lists, trees, and hash based containers. The standard algorithms are explained in detail while going through a series of examples. We will develop new data structures ourselves based on the knowledge of the existing containers in the C++ standard library.

PREREQUISITES

- Credit or credit in MATH 1550 or registration in MATH 1435, credit in CSC 1253
- There are no other required prerequisites, except for willingness to learn and some interest in programming
OFFICE HOURS

While I don’t have “office hours” in the conventional sense, I will stay around after lectures to answer questions. If this doesn’t work out, send me an email (hkaiser@cct.lsu.edu) and I can try to meet with you at another, more convenient time.

TA | TBA | TBA

LECTURES

Tuesdays and Thursdays, 10:30am to 11:50am R203 Tureaud Hall

IMPORTANT DATES

- Midterm exam: March 5
- Mardi Gras holidays: February 16-17
- Spring break: April 6-10
- Final exam: May 6, 3:00pm - 5:00pm in R203 Tureaud Hall

Both exams are comprehensive.

READING

We will use material from the following book: Andrew Koenig, Barbara Moo, Accelerated C++: Practical Programming by Example (ISBN 9780201703535). I suggest you acquire a copy for yourself. This book is not a conventional textbook and does not have any gentle introduction to C++. If you feel you need some introductory textbook for C++, please consider Bjarne Stroustrup, Programming - Principles and Practice Using C++ 2nd Ed. (ISBN 978-0-321-99278-9).

Complementing this and the lectures, I recommend reading some of the following books, each of which is outstanding in its own right:

- C++ Primer, 5th Edition by Stanley Lippman, Josée Lajoie, Barbara E. Moo. This books gives a comprehensive introduction into C++11, it’s definitely one of the best text books for this.
- Effective C++, More Effective C++, and Effective STL by Scott Meyers. These books are incredibly useful and will change the way you think and program in C++. The books in the Effective C++ series are collections of useful pieces of advice, so you can quickly navigate to relevant sections. While Meyers' books are targeted at audiences with a solid understanding of core C++ language features, they should still be quite useful in the second half of the course.
- The Design and Evolution of C++ by Bjarne Stroustrup. This most excellent book by the creator of C++ provides insight into the design decisions and overarching philosophy of C++. If you’re interested in how the language came to be, this is the definitive resource.

In Addition to the above books, I strongly recommend picking up a C++ language and library reference book for use in this class. Stroustrup’s The C++ Programming Language is an excellent choice.
Other resources:

- Mailing list: csc1254@cct.lsu.edu.

Please don’t hesitate to ask questions related to the course by sending me emails: hkaiser@cct.lsu.edu.

PROJECTS, HOMEWORK AND QUIZZES

There will be up to seven homework assignments, which will be due at the beginning of class on the due date. A penalty of 15% will be assessed for each day homework is late up to a maximum of 30%. For your final grade I will take into account all homework assignment results.

There will be one programming assignment (project) which will be due at midnight (11:59pm Central Time) of the due date. A penalty of 10% will be assessed for each day the project is late up to a maximum of 30% after which the project will not be accepted.

Occasionally we will have short quizzes (2-3 questions in 5 minutes) at the beginning of a lecture. These are mostly about things discussed in one of the previous lectures and shouldn’t require any special preparation – if you follow the lectures, that is.

You will be responsible for regularly following any course related information, such as homework assignments and due date announcements on the course website (http://www.cct.lsu.edu/~hkaiser/spring_2015/csc1254.html).

GRADING

- Quizzes and homework 30%
- Project 20%
- Midterm exam 20%
- Final exam 30%

Overall, in the end of the semester 90% of all points or more will give you an ‘A’, 80% or more a ‘B’, 70% or more a ‘C’, and 60% or more results in a ‘D’. Below that you’ll fail the course, but I’m sure that will not happen to anyone.

TOPICS

The following list indicates roughly how much time we will spend on each topic:

<table>
<thead>
<tr>
<th>No. of Weeks</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>C++ review, strings, vectors, lists, stacks, queues</td>
</tr>
<tr>
<td>6-11</td>
<td>Library algorithms and data structures, generic functions</td>
</tr>
<tr>
<td>12-15</td>
<td>Memory management, inheritance, dynamic binding</td>
</tr>
</tbody>
</table>

There is a full lecture calendar available on the course webpage outlining the topics in more detail. There is also some flexibility in shortening some of these topics and adding other advanced topics.
COURSE POLICY

GRADING

It is course policy that whoever graded something will be responsible for handling grading disputes. I will grade the midterm exam and the final exam. The grader will grade the homework and the project. Grades become final one week after homework or exam is handed back. This should leave ample time to resolve grading disputes.

HOMEWORK STANDARDS

All homework and the project have to be submitted using the electronic submission mechanism as outlined in an appendix to the first homework assignment. All work submitted must carry the student’s name and must be reasonably neat and well organized. Any work that cannot be read easily will score zero points. A reasonable standard of English expression and grammar is also required. The same requirements apply to exams. Additional requirements may apply for any of the separate assignments and will be outlined in the corresponding descriptions.

PROGRAMMING STANDARDS

The algorithm used must be essentially correct. Obviously, the program should (compile and) run. Because of the complexity of the programs, no credit can be given for a program that doesn’t run. If a program core dumps, only partial credit will be given.

HONESTY

I will treat you as professionals, and you should plan on conducting yourself as such. This course presents many important concepts you will need throughout your career as a computing professional, so it is important that each student do all the assignments and projects and learn the material.

You are free to discuss homework assignments and the project with others. In fact, I encourage you to do so. However, the solutions you submit are to be developed by yourself. Cheating is a very serious offense and will not be tolerated. Supplying others with homework solutions or material is also forbidden. The policy is that the supplier and the receiver of information will both be dealt with in accordance with and as outlined by the LSU Code of Student Conduct (http://saa.lsu.edu/Code%20of%20Student%20Conduct%20August%202009.pdf).

LAPTOPS

I’m not generally opposed to you using laptops during the lectures. Many students either take notes or try to find additional references helping them to follow the material. But unfortunately the experience shows that laptops may pose a major distraction not only for those using them but as well for the students sitting around. Please understand that I may have to restrict the use of laptops in class at some point.