COMP322: Introduction to C++ - Winter 2017
Monday 13:35-14:25, Burnside 1B45

1 General Information

This one-credit course will provide a very basic introduction to the C++ programming language.

This course will use mycourses. All information such as announcements, lecture notes, assignments, etc will be posted there. There also will be a course discussion board where you may ask questions of other students, TAs, and the instructor.

The instructor for this course is Dan Pomerantz. I can be reached most efficiently by email at `dpomer@cs.mcgill.ca`. My office is McConnell 328.

Office hours: Monday from 14:30:15:30 (after class). I'll also be available by appointment as needed.

The teaching assistant information will be announced on mycourses when it is finalized.

2 Description

C++ is a popular object-oriented programming language, originally developed at Bell Labs by Bjarne Stroustrup from 1979-1983. C++ is descended from the C programming language (Kernighan and Ritchie, 1978) and is an ancestor of the Java language (Gosling, Joy and Steele 1996). This course will cover the essential features of the C++ programming language. We will focus on the features that distinguish C++ from Java (pointers, memory allocation, templates) and C (classes, operator overloading, namespaces, exceptions). We will touch on the some of the services provided by the C++ standard library. Given the limited time available in lecture, we cannot attempt to cover every single concept of C++. If there is a particular topic you would like addressed, please reach out to me and we can see if it's possible to include in lectures or at least find external resources.

3 Prerequisites

This course requires that you have already completed at least COMP 250 AND either COMP 206 or COMP 208. As much of the material will discuss contrast C++with C and Java, you should be very familiar with both C and Java.

If you have not met all of these prerequisites, you should not take this course. If you are concerned about this or worried about your knowledge of C or Java, please discuss with your instructor.

While we do not have time to turn this into a full object-oriented (OO) programming course, we will attempt to explain OO principles at least as they apply to C++. This will often involve contrasting with Java, which is also an OOP language.

4 Goals of the course

At the end of this course, you should:

1. Understand the differences between C++, Java, and C.

2. Properly manage memory in a programming environment without garbage collection.

3. Know how to use some of the features of C++such as the standard libraries and iterators.

4. Understand the basics of object oriented programming.

5. Know some of the many tricks of C++and further to have an idea of *when* to apply these tricks and when to stick with the basics.

6. Have a richer understanding of programming based on seeing a different way to do things than in Java.

5   Course Resources

All course resources will be found on mycourses. It is your responsibility to check both of them in order to get up to date information on assignments, quizzes, lectures, announcements, etc.

All assignments will be submitted on my courses. In addition, there will be discussion boards on my courses which will be monitored by the instructor and TAs. You are encouraged to post questions here as then other students can help as well and share their knowledge. You may not post your code on my courses, but you may post small snippets of code (e.g. 10 or so lines) in order to clarify why something isn't working. When in doubt, err on the side of too few lines of code and perhaps your instructor will request you post more to clarify.

6   Textbook

There is no formal textbook for the class. However, I recommend that you obtain one of the standard reference texts on C++. One such book is *The C++ Programming Language: 3rd Edition* by Bjarne Stroustrup.

Another highly recommended book is *Accelerated C++* by Andrew Koenig and Barbara Moo. You can purchase this at `http://www.amazon.ca/Accelerated-C-Practical-Programming-Example/dp/020170353X`.

7   Evaluation

There will be three homework assignments and two quizzes.

You will have roughly two or three weeks to complete each assignment (this may vary slightly). As the purpose of the class is to familiarize you with the C++language, assignments will consist mostly of programming problems. An important component of any programming work is documentation, therefore you should use comments throughout the code to explain the reasoning behind your design choices. Both the quality and functionality of the code and the completeness and accuracy of your comments will be taken into account in the grading of assignments.

You are welcome to discuss assignments with your fellow students, but all submissions must reflect individual work.

Assignments *must* be submitted electronically via mycourses, by the due date. Late assignments will be penalized by 10 percent per day up to a maximum of 2 days. For example, an assignment received 25 hours late will be eligible for at most 80% of the possible score. An assignment received 50 hours late will not be accepted except in extenuating circumstances (e.g. illness).

The quizzes will be given in class, and will consist of roughly 10 short answer or multiple-choice questions. On quiz days, assume you will have about half the class time to complete the quiz.

Each assignment or quiz will be worth 20% of the total final grade.

In accord with McGill University's Charter of Students' Rights, students in this course have the right to submit in English or in French any written work that is to be graded.

8   Approximate Schedule

Note that this is a *tentative* schedule only. It may be revised as the semester progresses. Note that the dates for the assignments in particular may change and won't necessarily be due on Mondays.

1. 09 Jan - Course introduction, some basics of C++

2. 16 Jan - Basic language features

3. 23 Jan - Pointers and references (Assignment 1 out)

4. 30 Jan - Memory management

5. 06 Feb - Input/output using the Standard Library (Assignment 1 due, Assignment 2 out)

6. 13 Feb - Standard library basics

7. 20 Feb - Quiz 1

8. 27 Feb - **Reading week**

9. 06 Mar - Classes in C++, Overloading, Inheritance (Assignment 2 due, Assignment 3 out)

10. 13 Mar - Inheritance continued

11. 20 Mar - Exceptions

12. 27 Mar - Templates

13. 03 Apr - More on standard libraries (Assignment 3 due)

14. 10 Apr - Quiz 2

9   Academic Integrity

McGill University values academic integrity. Therefore all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures (see www.mcgill.ca/integrity for more information).

L'université McGill attache une haute importance à l'honnêteté académique. Il incombe par conséquent à tous les étudiants de comprendre ce que l'on entend par tricherie, plagiat et autres infractions académiques, ainsi que les conséquences que peuvent avoir de telles actions, selon le Code de conduite de l'étudiant et des procédures disciplinaires (pour de plus amples renseignements, veuillez consulter le site www.mcgill.ca/integrity).