

# Advanced C++ Programming

CSC 218 Fall 2004

September 16, 2004

## 1 Instructor, TA, and Venue

Lectures will take place in Nichols 315, Mondays and Wednesdays at 4:30 pm. Roughly one third of the lecture time will be however spent in Johnson 118, where you will have a chance to actually do C++ programming, not just hear about it.

The instructor is Stefan Bruda (email: [bruda@cs.ubishops.ca](mailto:bruda@cs.ubishops.ca); telephone: 822-9600 ext. 2374; office: Johnson 114b). The TA is Petter Haggholm (email: [phaggholmOl@ubishops.ca](mailto:phaggholmOl@ubishops.ca)).

During office hours you can find me in either Johnson 114B or Johnson 118. Official office hours are Tuesdays from 6:15 pm to 9:15 pm and Wednesdays from 10 am to noon. I am generally available outside these official hours (maybe with some advance notice), feel free to bother me each time you get hold of me, or write to me to establish an appointment. In addition, questions will be answered by email. In general, outside classes and meetings face to face, email shall be the primary mean of communication—I am most easily reached by email should you have any questions or concerns outside the classroom, and once I send you an email I shall hereby consider that you read and acknowledged its content.

Petter's office hours will be announced shortly.

### 1.1 Web Coordinates

The official Web page of the course is <http://turing.ubishops.ca/home/csc218>. All the information related to the course will appear here and may not be available in any other place. In particular, assignment handouts will be posted here only; I will not hand out hardcopies in class. Always consult the course Web page for updates to this outline, assignments, lecture notes, and other resources.

## 2 Course Outline

We talk about the C++ programming language, which has become an industry standard for application programming (system programming is not that far off, the standard here being C++'s cousin C). Since we are at it, we shall also talk about programming style and good programming practices.

To make learning C++ fun (and more useful), all but the most trivial programming done throughout the course will actually make some sense in other respects than merely helping you to learn the language. Specifically, we shall consider implementations of algorithms (such as sorting, parsing, compression) and data structures (stacks, queues, trees, graphs, etc.) that are common place in practice.

Time permitting, we shall also discuss abstract data types (together with their implementation) and more advanced algorithms.

We thus base the course on the following rough (i.e., subject to change) outline. Note that this outline is not necessarily in chronological order; for instance, the programming process shall be discussed as soon as you have enough base knowledge to be able to understand it.

Introduction, rules of the game, etc	1 week
Elementary C++ programming	2 weeks
Arrays, pointers, and structures	3 weeks
The programming process (specification, testing, debugging, maintenance)	1 week
Advanced types, classes	2 weeks
The C++ preprocessor	1 week
Advanced classes, templates	2 weeks
Advanced topics	1 week

### 3 Grading

Grading will be based on a number of tokens. Tokens are earned by doing assignments and quizzes based on previous assignments. There is no final or mid-term examination.

You will be given one programming assignment about every two weeks. Each assignment is worth a strictly positive number  $n$  of tokens (to be established when the assignment is published). You can receive any number of tokens between 0 and  $n$ . Exceptionally good solutions will also receive 10% “bonus” marks. The degree of difficulty of the assignments (and hence the number of tokens awarded for them) will increase steadily. In particular, toward the end of the term there will be a larger assignment (aka project).

Unless stated otherwise in the assignment handout, you are allowed to collaborate in doing your assignments in groups of no more than two; all collaborators must be named and must be currently taking the course.

Upon completion of selected assignments, you will have to do an open-book, in-class quiz. The questions in the quiz will be related to the just completed assignment. Each quiz will be no more than 30 minutes in length (often much shorter) and is worth one or two tokens. Quizzes will be usually given at the end of the class. They will not be announced in advance. You can get any number of tokens between zero and the maximum (one or two) for a quiz; there are no bonuses.

Exactly all the tokens will contribute equally to your final grade for the course.

#### 3.1 Special needs

Should you require special accommodations because of a disability, please come and discuss this with me at the beginning of the term. I will not take into consideration requests made on or after the day in which the first assignment is released. You must also contact the Counselling Resource Centre to obtain authorization for any special arrangements.

### 4 Textbook

Robert Lafore, *Object-Oriented Programming in C++*, fourth edition, Sams, 2001. Available at the campus bookstore. ISBN 0-672-32308-7.

In general, any reasonable C++ textbook will do. In principle, such a textbook combined with my lectures should be enough—remember, this is a programming course, it is not really about memorizing texts. I will organize however my lectures around the mentioned text, so you may find convenient to have it at hand (it is quite cheap too).