C++ for Financial Mathematics

Dr John Armstrong

King's College London

June 30, 2017

Introduction

Course overview

An introduction to C++ with examples from finance.
These are the milestones we wish to achieve:

- Pricing a portfolio of derivatives.
- Modelling a market with multiple stocks.
- Writing a multi-thread pricer

Why learn C++?

For you:

- It will help you get a job

For banks:

- C++ is general purpose.
- C++ gives you direct access to processors and memory.
- C++ is designed for large programs.
- C++ is compatible with C.
- They have lots of C++ code already.

In this course

You will learn how to use C++ to achieve the following:

- Access computer memory directly.
- Take advantage of multiple processors.
- Write programs that are easy to test and maintain.
- Use object-oriented techniques to write large programs that are still easy to understand.

How important is performance?

Example

A student wishes to price a derivative for their MSc dissertation. They estimate that the program will take 10 minutes to run if they write it in MATLAB but will only take 2 minutes to run on a quad-core computer if they write it in C++. Which language should they use?

Pricing a portfolio of derivatives

- How many stock exchanges can you name?
- How many types of equity derivative can you think of?
- How many types of derivative can you name?
- What statistics might you report on a particular position?

The problem is not one of mathematics, it is one of scale and complexity.

Working with teams

Problem

How do you write software so that no individual has to understand everything that is going on?

Problem

How do you write software so that a team of hundreds can work on the software at the same time without getting in a mess?

Problem

How do you write code that is easy for others to understand?

Correctness

Problem

How do you write code that doesn't contain bugs? How do you ensure that there are no bugs in the code written by a team of hundreds?

Problem

Given that you probably can't guarantee that there are no bugs, how do you ensure that the effects of a bug are not too harmful?

Extendability

Problem

How do you write code that can be extended easily and rapidly?

Problem

How do you ensure that no bugs have crept into the latest version of your code, given that you plan to release a new version almost daily?

Problem

How do you even release new code, when all the software has to keep running 24×7*?*

Scalability

Problem

How can you ensure that your software will continue to work with exponentially increasing data volumes?

Exercise

Which is more useful:

(A) A computer program that computes the correct answer in an hour.

(B) A program that computes an incorrect answer in 8 seconds?

Summary

- You will learn C++
- You will learn about *software quality*.
- We will consider pricing a portfolio of derivatives to illustrate *scalability and maintainability*.