DATASIM EDUCATION BV

MoneyScience Financial Training www.moneyscience.com

2 – 4 June, 2010
18 – 20 October, 2010
Liverpool Street, London, UK

'Early Bird' Discounts Available

New, Updated Course and Materials

Daniel J. Duffy presents

Advanced C++ for Computational Finance

The goal of this **three-day intensive hands-on course** is to learn those advanced features in C+ that are of direct relevance to writing and extending application for quantitative and computational finance. The course uses the object-oriented and generic (templates) programming models (OOP, GP) in combination with design patterns and the STL and boost libraries to allow you to create robust and flexible applications. We develop the contents of the course by discussing important C++ language, using OOP and GP models to write clean and effective code. We also discuss how to improve the performance of your application. In all cases, the examples and test cases are based on finance experience.

This is one of the few courses (in our opinion) that focuses on the application of C++ to quantitative and computational finance. It is a practical course for practitioners.

To participate in this course, you need to bring your own laptop computer with a C++ compiler (ideally Microsoft's Visual Studio or GNU GCC for example)

The percentage theory/practice is 70/30.

About the speaker

Daniel J. Duffy has BA (Mod), MSc and PhD degrees, all of which in mathematics and numerical analysis. He has been working with numerical methods on finance, industry and engineering since 1979. He has written four books on numerical methods and C++ for quantitative finance and he has developed a number of new schemes for this field as well as more than 20 years of training experience.

What do you learn?

In this course we introduce state-of-the-art design and programming techniques in C++ and their application to Computational Finance. In particular, the following topics are discussed in detail:

- Advanced C++ syntax and its application
- Template classes and the Standard Template Library (STL)
- Combining the object-oriented and generic programming paradigms
- The famous Gamma (GOF) design patterns applied to QF
- Interfacing to Excel: COM Add-ins
- Creating applications: Monte Carlo, Finite Difference and lattice methods

What do you receive?

As attendee you receive a full set of slides, C++ source code and a copy of Daniel Duffy's book "Financial Instrument Pricing using C++" (Wiley 2004), including CD with C++ code. In short, you will receive what is needed to start developing your own applications. The price includes coffee, tea, lunch and refreshments.

What have previous delegates said?

- ***"Very good style and knowledge was far above the norm"***
- ***"Excellent hands-on teaching"***
- ***"Good balance of C++ and finance, theory and practice"***
- ***"The book I wish I had had when I first started studying C++"***
- ***"The best course I ever attended (Vienna)"***

Prerequisites

We assume that the student has experience of C++. This is not a beginners course and we assume you know what constructors, destructors and operator overloading are in C++ and how memory management works (see modules 1-3 of the Distance Learning course).

Who should attend?

This course has been developed for financial professionals who design and implement pricing and hedging models in C++ and Excel. The course introduces and elaborates on how to apply C++ to creating flexible and reliable applications in Quantitative Finance using the most modern software design techniques. There is ample room for questions on your own specific applications as well as hands-on programming sessions. It is assumed that the attendees have some working knowledge of C++ and have developed applications or prototype applications in that language.

DATASIM EDUCATION BV

MoneyScience Financial Training www.moneyscience.com

Course contents

Day 1: Advanced Object-Oriented and Generic C++

Memory Management, Issues
- Static, stack and heap memory management
- Wild pointers, dangling pointers, double free bugs, memory leaks
- Using object factories to control object lifecycle
- Single object and object array allocation and deallocation

Memory Management, Solutions
- The pimpl idiom; STL auto_ptr
- Using Builder pattern to coordinate object lifetime
- Overview of Boost smart pointers
- Scoped and shared pointers
- Casting and Run-Time Type Information (RTTI)
- Static and dynamic casting overview
- dynamic_cast and static_cast
- Exception handling issues
- Boost lexical_cast

Polymorphism Functions
- Dynamic polymorphism: virtual and pure virtual functions
- Polymorphism and algorithms in computational finance
- Polymorphism, inheritance and composition, what's best?
- Performance issues with dynamic polymorphism

Static Polymorphism
- Curiously Recurring Template Pattern (CRTP)
- Where does static_cast fit it?
- Where to use CRTP
- CRTP and performance improvements

Advanced Generic Programming (GP) in C++
- Template classes and template functions
- Partial specialization
- Default parameters, template template parameters
- Template member functions
- Nested templates and data structures

Combining OOP and GP
- Inheritance scenarios
- Composition; combining composition with inheritance
- Policy-based design and traits classes
- When OOP and when GP?

Day 2: STL, Boost Libraries and Design

STL Containers
- Sequential and associative containers
- Lists, vector and queues
- Maps, sets and multimaps
- Modelling option data with maps and Property Sets

STL Algorithms and Iterators
- Mutating and non-mutating algorithms
- Searching and sorting
- Inserting and removing data

Using STL in Applications
- Using STL with class adapters
- STL-compatible data structures
- Complexity analysis and performance tests

Function Pointers and Function Objects
- C function pointers: advantages and disadvantages
- Function objects and their applications
- Functions in boost; binding
- Performance issues

String Algorithms
- Regular expressions
- Trimming and conversion
- Find and replace
- Find iterator
- Join and split

Random Number Library
- Concepts
- Random number variate generators
- Random number library distributions
- Applications

uBLAS (Basic Linear Algebra System)
- Vectors, matrices and their operations
- Patterned matrices (sparse, triangular, ..)
- Expression templates to improve performance
- Applications of uBLAS classes

Essential Patterns: Creational
- Factory method and abstract factory
- Creating complex objects using Builder
- Singleton<T>

Essential Patterns: Structural
- Composite and nested objects
- Bridge and implementation-independence
- Extending object structure with Decorator

Essential Patterns: Behavioural
- Strategy and algorithms
- Extending class functionality with Visitor
- Template method pattern and customisable frameworks

Day 3: Design and Applications to Computational Finance

Multithreading: Theory
- What is a thread? Thread lifecycle
- Shared data models
- Thread synchronization and notification
- Speedup and race conditions

Multithreading: OpenMP and boost Thread
- Overview of functionality
- Creating threads
- Speedup, accuracy and robustness

Excel-C++ Integration
- An overview of Excel add-ins
- Creating Automation add-ins and worksheet functions
- Creating COM add-ins: The steps using ATL projects

The Monte Carlo Method
- Description of the problem
- Creating a software framework for a MC engine
- Using design patterns to create flexible MC systems
- Plain options, Asians and barriers

The Finite Difference Method
- A quick introduction to FDM
- C++ classes for a FDM solver
- Explicit and implicit schemes (Crank-Nicolson, Euler, ADE)
- Presentation in Excel

Improving Application Performance
- Call by reference versus call by value
- Appropriate use of virtual functions
- Function objects versus function pointers
- Preventing unnecessary temporary object creation

Loop Optimisation
- Loop interchange
- Loop fission and fusion
- Making loops multi-threaded (OpenMP)
- Application profiling; determining a program's serial fraction

Note:
To participate in this course, you need to bring your own laptop computer with a C++ compiler (ideally Microsoft's Visual Studio or GNU GCC for example)

Further Information
Dates: 3 Days
2 – 4 June, 2010
18 – 20 October, 2010

Venue: Liverpool Street, London, UK
Cost: £3000 exc. VAT

Registration
Please Return Application form to:

Financial Training
MoneyScience Ltd
4 St Pauls Road
Bristol
BS8 1LT
UK

Enquiries:

Jacob Bettany

Tel: +44 (0) 117 923 8851
Email: jacob@moneyscience.com

Advanced C++ for Computational Finance - Registration Form

This form relates to participation on ☐ June 2 - June 4 ☐ October 18-20

Title:	
First Name:	
Last Name:	
E-mail address:	
Participant's First Name (If different from above):	
Participant's Last Name (If different from above):	
Participant's E-mail address (If different from above):	
Position:	
Telephone:	
Company Name:	
Tax Registration No:	
VAT Identification No:	
Department:	
Address:	
Town:	
State:	
Postal Code:	
Country:	
How did you hear about this course?	
Comment or requirements	

Please Return Completed Application to:

SIGNED..

Financial Training,
MoneyScience,
4 St Pauls Road,
Bristol, BS8 1LT, UK

DATE..

or Sign, Scan and Return to jacob@moneyscience.com

☐ **I have read and understand the Terms and Conditions on the page following.**

Cancellation
If a person which has registered for this course is unable to attend for whatever reasons, a substitute representative may be appointed to participate in advance. Cancellation will be accepted only in written form no later than twenty calendar days prior to the date of the course while a cancellation fee in the amount of 10% of the price or block price will be charged. Cancellation of the participation within a period of less than twenty calendar days prior to the date of the course is not possible for the technical and the organisational reasons. MoneyScience reserve the right at all times to cancel participation or cancel the entire seminar for whatever unspecified reasons, including possible force majeur. In this case, the price will be refunded in full to the applying participant.

DATASIM EDUCATION BV

MoneyScience Financial Training www.moneyscience.com

Terms and Conditions:

MoneyScience Training Events are English-language seminars, training sessions and practical workshops delivered by expert, qualified tutors. Our objective is to provide industry professionals with advanced financial know-how and up-to-date analytical methods and skills.

Tuition Language
MoneyScience Events are held in English and all relevant manuals, training software etc. are also provided in English. In order to benefit from participation, a good working knowledge of English, including common financial phrases and related terminology, is required.

Registration
Clients who decided to participate, should submit their registration by mail to us in advance. Subject to availability, the participant will then receive a confirmation of participation. The number of participants is strictly limited to preserve an effective and focused learning environment.

Seminar Prices
The quoted seminar prices are per person and include all course material, lunches and refreshments, The price does not include hotel accommodation. Quoted prices are exclusive of VAT.

Discounts
Bulk discounts are offered when submitting an application for the participation of more than one person. Please contact jacob@moneyscience.com for further details.

Invoicing and Payment
An invoice for the seminar price will be sent to the participants on receipt of their Registration Form and should be paid within days. Full payment of the invoice must be made before the start of the course as a precondition of participation.

Terms of Cancellation
If for whatever reasons a registered participant is unable to attend, a substitute delegate may be appointed to participate instead. For cancellations received 20 days or more before the beginning of the seminar, a 10 % cancellation fee of the full price will be invoiced i.e. 90 % of the price is refunded. For cancellations received less than 20 days prior to the beginning of the seminar, the full price is payable i.e. no refund will be provided. All cancellations must be in writing. MoneyScience reserves the right to cancel the individual participation or cancel the entire seminar or part of it for whatever unspecified reasons, including possible force majeure. In this case, the price paid will be refunded in full or in part, accordingly.