

Fall 2018
FRE-GY 6883 Financial Computing
Song Tang, st290@nyu.edu, 646-283-4578

Overview:

This course covers programming applications to financial engineering, including C++ and Java and the various common development environments for them. Topics include structured and object-oriented programming in C++ with applications to binomial options pricing, multi-threaded programming in Java with applications to risk measurement tools, market data access libraries with applications to historical financial data series retrieval and management, and other advanced programming concepts important for financial engineering such as numerical techniques, trading systems, and large-scale software design.

Schedule of Classes:

Week	Topics
1	Topic 1: Financial Computation in C/C++ Part 1
2	Topic 1: Financial Computation in C/C++ Part 2
3	Topic 1: Financial Computation in C/C++ Part 3
4	Topic 1: Financial Computation in C/C++ Part 4
5	Topic 2: Financial Applications Using Data Structures and Class Templates – Part 1
6	Topic 2: Financial Applications Using Data Structures and Class Templates – Part 2
7	Topic 3: Implementation of Non-linear Solvers in C++
8	Topic 4: Monte Carlo Methods – Part 1
9	Topic 4: Monte Carlo Methods – Part 2
10	Midterm Exam
11	Topic 5: Working with Historical and Market Data
12	Topic 6: Trading Systems & Large Scale Financial System Design
13	Topic 7: Financial Numerical Recipes in C++
14	Topic 8: Financial Computation in Java
15	Team Project Presentation and Demonstration

Assessment:

Students will do extensive coding for both their homework assignments and the final project. Homework assignments are requested to complete independently. Most of homework assignments are closely related to the topics and programs discussed in classes. The final project will be done in group of 4 students. The final project will be given to students around midterm week, so students will have 5-6 weeks working on their projects. The last class will be the project presentation and program demonstration.

The assessment will be done as the following:

- Quizzes, 10%
- Homework Assignments, 30%
- Midterm Exam, 30%
- Group Project, 30%

Textbooks:

- (1) Numerical Methods in Finance with C++ (Mastering Mathematical Finance), by Maciej J. Capinski and Tomasz Zastawniak, Cambridge University Press, 2012, ISBN-10: 0521177162
- (2) Introduction to C++ for Financial Engineers: An Object-Oriented Approach (The Wiley Finance Series), by Daniel J. Duffy, Wiley, 2006, ISBN-10: 0470015381

Development Environment:

- (1) C++ Integrated Development Environment
 - a. PC: Microsoft Visual Studio Community 2015 or Visual Studio Community 2017, a free version of Microsoft Visual Studio.
 - b. MAC: Xcode, should be preinstalled, otherwise, download from Apple Web site.
- (2) Java Development Environment, students will use Dr Java or Eclipse IDE for Java developers. Both are free IDE environment. Dr Java is a lightweight development environment for writing Java programs. It is designed primarily for students, providing an intuitive interface and the ability to interactively evaluate Java code.

Course Topics:

Financial Computation in C/C++

- The Structured Programming

Structured programming is a technique which arose from the analysis of the flow control structures which underlie all computer programs. It is possible to construct any flow control structure from three basic structures: sequential, conditional and iterative. Students will learn how to price European options via the binomial model.

Topics covered in this section:

 - Principle of Structured Programming
 - Program Shell
 - Entering Data
 - Functions
 - Separate Compilation
 - Cox-Ross-Rubinstein (CRR) Pricer
 - Pointers and Function Pointers.
- Object-Oriented Programming in C++

In this unit students will gain a basic understanding of object oriented programming using C++ and the design of a C++ application based on what they did in structured programming. In other words, they will recast their option pricer in the style of object-oriented programming. Their classes will reflect the relationships between real entities, namely the binomial model and European options of various kinds.

Topics covered in this section:

 - Our First Class
 - Inheritance
 - Virtual Functions
 - Recast the Option Pricer

Financial Applications Using Data Structures and Class Templates

This unit will introduce students advanced object-oriented programming techniques and demonstrate how these techniques could be applied to solve complicated problems in quantitative finance. Students will learn how to implement American options by using C++ data structures and class templates.

Topics covered in this unit:

- Multiple Inheritance
- Virtual Inheritance
- Class Templates
- Computing Option Price via Black-Scholes Formula

Implementation of Non-linear Solvers in C++

This unit will use C++ function pointers, virtual functions, function templates, and design patterns to implement nonlinear solvers.

Topics covered in this unit:

- Bisection Method
- Newton-Raphson Method
- Function Pointers
- Virtual Functions
- Function Templates
- Computing Implied Volatility

Monte Carlo Methods

This unit cover basics of Monte Carlo simulations for path-dependent options, with emphasis on practical issues such error analysis, variance reduction and algorithm updates.

Topics covered in this unit:

- Path-dependent Options
- Valuation
- Pricing Error
- Greek Parameters
- Variance Reduction
- Path-dependent Basket Options

Working with Historical and Market Data

This unit will let students gain an understanding of financial market data processing. It will teach student how to integrate Excl in a C++ application, and analyze historical market data from Access database for trading in a C++ program. In addition, it will demonstrate to students how to fetch historical market data from Yahoo Finance using libcurl in C++.

Topics covered in this unit:

- Access Market Data using libcurl in C++
 - Handle the Easy libcurl.
 - Configure Microsoft Visual Studio for Using LibCurl
 - Fetch historical stock data from Yahoo Finance

- Communication between C++ and Excel
 - Object Model in Excel
 - Technical Details of C++ Interfacing to Excel
 - Two Main Function Entities of Excel Driver Package
 - Singleton Design Pattern used by ExcelDriver
 - Option Values and Sensitivities in Excel

Trading Systems & Large Scale Financial System Design

This unit is to introduce students the concepts, terminology and code structures required to develop trading applications, as well as high frequency trading and dark pool. In addition, this unit will show students a comprehensive top-down approach to the logical design of individual components for large-scale financial projects.

Topics covered in this unit:

- Automated Trading Systems
- Single-Threaded, Technical Analysis System
- Multithreaded, Statistical Arbitrage System
- High Frequency Trading
 - US Equity Share Volume by Market Participant
 - US Equity Trading by Market Participant
 - Challenges in High-Frequency Trading
 - HFT Technologies
 - Dark Pools
- Software Development Cycle
 - Ad-hoc Development
 - Lifecycle Stages
 - Benefits and Limitations of Lifecycle Models
 - Code-and-fix Model
 - Waterfall Model
 - Spiral model – Risk Oriented
 - Staged Delivery Model
 - Evolutionary Prototyping Model
 - Design-to-Schedule
- Trade System Development Cycle
 - Key Ideas in K/V Software Methodology
 - Advantages of K/V Development
 - K/V Trading System Development Stages

Financial Numerical Recipes in C++

This unit will give students a comprehensive demonstration of C++ implementation of a variety of numerical methods for bond pricing and term structure.

Topics covered in this unit:

- Bond Pricing
 - Present Value
 - Bond Price

- Yield to Maturity
- Duration
- Measuring Bond Sensitivity to Interest Rate Changes
- Convexity of a Bond
- Continuously Compounded Interest
- Term Structure of Interest Rate
 - The Term Structure as an Class
 - Flat Term Structure
 - Linear Interpolation
 - Interpolated Term Structure Class
 - General Term Structure & Continuous Compounding
- Term Structure Models
 - The Nelson Siegel term structure approximation
 - Cox Ingersoll Ross
- Futures Algorithms
 - Pricing of futures contract

Financial Computation in Java

This unit will introduce students to Java programming environment and present the main characteristics of Java, based on their knowledge of C++. Students will rewrite the Option Pricer in Java.

Topics covered in this section:

- The Java Development Environment
- The Difference between C++ and Java
- The Characteristics of Java Language
- Implement CCR Pricer in Java

Advanced Java Programming for Financial Issues (Optional based class progress)

This unit will introduce students to several advanced topics in Java programming and show how use Java multi-thread application for processing large amount of financial data efficiently and how to Java networking programming to access market data from Internet.

Topics covered in this section:

- Networking Programming
- Multi-threading Programming
- The Event Driven Programming Paradigm
- Applets & Programming Graphics Interface in Java

Course Team Projects:

Students are required to do class projects in groups of 5 members. Groups once formed cannot be changed midway through the project. The team lead is responsible to facilitate the planning of the project, and the entire team will plan the project under the guidance of your team leader. Planning involves identifying what should be done (tasks), who should do it (resources), when tasks should be done (time frames) and how tasks are best sequenced (dependencies).

Each team will submit project report and source codes tar/zipped via email before the deadline. The project reports should include a brief executive summary, the project design approach, design choices and implementation specifics. All the teams are requested to present and demonstrate their projects.

The details of the team project will be announced and posted in our course site before midterm exam.