MF 703

Programming for Math Finance Midterm Exam

Professor Christopher Kelliher

November 5, 2020

Name:

BU ID:

Signature:

# Problem 1 (7 points) SQL Programming

Consider the following table design structure:

**instrument info** instrument id (int) ticker (char(5)) sector id (int)

shares outstanding (numeric)

**instrument prices** instrument id (int) ticker (char(5))

quote date (datetime) close (numeric)

open (numeric) low (numeric)

daily return (numeric)

# sectors

sector id (int)

sector name (char(50))

Table names are in bold, columns are listed in each line and data types are in paren- theses.

1. (2 Points) Write a query using only the **instrument prices** table that computes the maximum closing price between two dates for a given ticker.
2. (1 point) Which columns are available to you to join the **instrument info** and

**instrument prices** tables? Which column is preferable and why?

1. (2 points) Write a query that selects the current closing price and shares out- standing from the **instrument info** and **instrument prices** tables for all in- struments. Return data for all rows present in the **instrument info** table whether or not there are matching prices. Rows with no price in the **instru- ment prices** table should return a price of 0.0.
2. (2 points) Write a query that computes the total market value (shares outstand- ing times market price) of each sector returned in order of largest to smallest

sectors. Exclude sectors that have a market value less than a given value.

# Problem 2 (10 points) Data Cleaning & Algorithms

1. (2.5 points) You are given a data set of sector ETF data which is missing mul- tiple year-long periods for several ETFs. Describe the best approach to handle this missing data and explain what impact filling the missing data has on the distribution of the data as well as the assumptions implicit in your method. Write a piece of code that will find all such gaps in the data and fill them using the method described.
2. (2.5 points) Using Python or C++, implement an insertion sort that uses a binary search. Discuss the strengths and weaknesses of this algorithm.
3. (2.5 points) What is the template method? What object-oriented programming concepts does it leverage? Using the template method, write a class structure that efficiently handles pricing Lookback and European options via simulation from the Black-Scholes model.
4. (2.5 points) Explain the difference between a Local Working Copy, a Local Repository and a Remote Repository in git. What commands would you use to check in your code to the Local and Remote Repositories respectively? Why might you check in your code to the Local Repository but not the Remote Repository?

# Problem 3 (10 points) Finance Applications

*Note that in this question you are not required or expected to produce any code or implement any models (although you are welcome to do this to check your solution). Instead you are expected to clearly explain why you believe these relationships should exist.*

1. (2.5 points) Consider a european call option with *S*0 = 100, *K* = 120, *T* = 0*.*25 and *r* = *q* = 0. If you were to model this option with a Black-Scholes model with *σ* = 0*.*25 and with a Bachelier model with *σ* = 25 which model would lead to a higher price? Why?
2. (2.5 points) Consider an investor who buys a fixed strike lookback put option and sells a European put option with the same strike, expiry and other char- acteristics. What are the conditions at expiry that lead to the best and worst P&L for this investor? Why?
3. (2.5 points) Suppose you are analyzing a particular asset and you notice that historically implied volatility has been significantly higher than realized volatility and you expect this to continue. Explain how you would take advantage of this as an investor and what the connection is between the structure you propose and realized vs. implied volatility.
4. (2.5 points) You regress current returns for the S & P 500 against the previous day’s returns and find a statistically significant, negative, coefficient. Describe what this phenomenon is and what you would do to take advantage of it. You repeat the same analysis for VIX and find a statistically significant, negative coefficient as well. Would be able to take advantage of it? Discuss the difference between the two and the different implications for investors of these coefficients.

# Problem 4 (15 points) Simulation Algorithms

Consider the SDE for an OU process:

*dSt* = *κ*(*θ − St*)*dt* + *σdWt*

You are asked to use this SDE to write a simulation algorithm to price an American up-and-out call option. Recall that the price of an American up-and-out call option is defined as:

*c*0 = E˜ *e−rT* (*ST − K*)+1*M*

*T*

*<B*

Where *MT* is the maximum value of the asset over the period and *B* is the barrier level.

1. (2 points) List the set of model parameters and describe their meaning. Explain how to simplify this model to the Bachelier model.
2. (2 points) Comment on whether you think this would be a good model for an equity index? How about a rates or volatility model? Why or why not?
3. (3.5 Points) Write a piece of code in Python or C++ that you can use to simulate a path from the given stochastic process.
4. (3.5 Points) Write a piece of code in Python or C++ that will define the price function of your exotic option, given a simulated path.
5. (2 points) Name the simulation parameters that you must choose and describe how you would set them optimally.
6. (2 points) Describe three unit tests that you would create to ensure that your model prices are correct.

# Problem 5 (10 points) Python Concepts

1. (2.5 points) You are given two pandas Series with historical closing price data for two equity indices. The equity indices are for different geographical regions and therefore may have different holidays. Write a Python function that merges the two data frames without losing data for any dates.
2. (2.5 points) Write a python function that calculates rolling overlapping annual- ized returns and rolling overlapping annualized volatilities for the merged data frame that you constructed above. Join the index rolling return and volatility series to the price DataFrame that you created above. This function should take the period length of the return and volatility calculations as an input.
3. (2.5 points) You are given an additional pandas Series identifying the market regime. Write a Python function that joins this column to the dataframe created above and calculates the conditional return and volatility by market regime. Make sure that your code is generic enough to handle different numbers of market regimes.
4. (2.5 points) A Fibonacci sequence is a sequence that starts with 0 and 1, and then each number equals the sum of the previous two numbers. That is:

Fib*n* = Fib*n−*1 + Fib*n−*2

Write a Python function that implements a Fibonacci sequence using a recursive function.

# Problem 6 (20 points) Object Oriented Programming in Python

1. (4 points) Using Python create an abstract base class named OptionsPricer. The abstract base class should contain a constructor, a member function named price, as well as any attributes you believe are appropriate.
2. (6 points) Create two derived classes that inherit from your OptionsPricer class, EuropeanOptionPricer and LookbackOptionPricer. These classes should assume that a simulated path for the underlying asset is provided to them.
3. (5 points) Create an analogous class structure for a stochastic process consisting of a base class and Bachelier and Black-Scholes derived classes. Each class in the hierarchy should have a simulation method that implements the appropriate SDE and returns a simulated path that can be read by the OptionsPricer class hierarchy.

In addition, implement the appropriate operator to compare whether the simu- lated paths in each instance of a class are the same, greater than, or less than.

1. (5 points) Write a generic global function that takes an option type and stochas- tic process, as well as any additional parameters you need (i.e. strike, expiry) and dynamically returns the price of the correct option using the desired model.

# Problem 7 (8 points) C++ Concepts

Consider the following piece of C++ code:

1 class Foo {

2 public :

3 Foo () {

4 std :: cout << " calling default constructor" << std :: endl;

5 }

6

7 Foo ( int bar\_ ){

8 bar = bar\_ ;

9 std :: cout << " calling constructor with bar" << std :: endl;

10 }

11

12 Foo ( const Foo & other ){

13 this -> bar = other. bar;

14 std :: cout << " calling copy constructor" << std :: endl;

15 }

16

17 int getBarValue ( int x){

18 x = bar

19 return ( bar)

20 }

21

22 int getBarRef( int& x){

23 x = bar

24 return ( bar)

25 }

26

27 static Foo make Foo ( int bar){ return Foo ( bar); }

28

29

30 };

int bar;

31

32 int main ( int argc , const char \* argv []) {

33 Foo f;

34 f. bar = 5;

35 std :: cout << f. bar << std :: endl;

36

37 Foo f1 ( f);

38 std :: cout << f1 . bar << std :: endl;

39

f1 . bar = 10;

std :: cout << f. bar << std :: endl; std :: cout << f1 . bar << std :: endl;

Foo f2 ( f. bar);

bar = f2 . getBarValue ( f. bar \*5); std :: cout << f2 . bar << std :: endl; std :: cout << bar << std :: endl;

bar2 = f2 . getBarRef( f2 . bar \*5); std :: cout << f2 . bar << std :: endl; std :: cout << bar2 << std :: endl;

Foo f3 = Foo :: make Foo ( f2 . getBar ); std :: cout << f3 . bar << std :: endl;

return 0;

}

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

1. (1 point) Explain the parameter argv. What is its type and how is it being passed to the main function? Write a piece of code that would access the second element in argv and convert it to a double.
2. (2.5 points) What will be the output of the main function above? Explain each line of output, what is causing it and which object it is coming from.
3. (1 points) Modify the code so that the member variable bar cannot be accessed directly but can still be through accessor functions.
4. (1 points) Write a piece of code that will overload the assignment operator in the Foo class. The overloaded operator should print ”calling assignment operator” in the function body and also set the value of bar.
5. (2.5 points) Consider the following piece of code using the Foo class:

1

Foo f; f. bar = -5

Foo f1 = f; shared\_ ptr < Foo > fPtr;

if ( fPtr ){

2

3

4

5

6

7

8

Foo fCond ();

}

Foo f2 (50);

Foo f3 = f2 ; f3 . bar = 100;

f2 = f;

9

10

11

12

13

14

15

16

What will be the output of this code be? Justify why / which object each printed line is coming from.

Add a line of code that will initialize fPtr calling the default constructor to Foo above the if statement. What impact will this have on the output?

# Problem 8 (20 points) Object Oriented Programming in C++

1. (5 points) Write a base class Foo that contains a default constructor, a destructor and two member functions func1 and func2. The two member functions do not need to take or return any parameters. Define func1 such that it will have dynamic binding in inherited classes and func2 such that it will have static binding.

Each function should print to the console what class and function they are being called from (i.e. the constructor should print calling Foo constructor, the destructor should print calling Foo destructor and func1 should print calling Foo func1)

1. (5 points) Write a derived class FooKid that inherits from the Foo class. It should override the func1 and func2 and define their own constructors and destructors. As with the Foo class, each function that you implement should print to the

console what class and function they are being called from.

1. (3 points) Suppose you want to store your Foo and FooKid classes in a hy- pothetical sortedList class which sorts the items as they are added. Describe what operators might need to be overriden for your class to work with this data structure and show you would implement these operators.
2. (7 points) Consider the following piece of code that leverages the Foo and FooKid classes which you should have written:

1

int main ( int argc , const char \* argv []) { std :: shared\_ ptr < Foo > foo1 ( new Foo ()); std :: shared\_ ptr < FooKid > foo2 ( new Foo Kid ());

foo1 -> func1 (); foo1 -> func2 ();

std :: shared\_ ptr < Foo > foo3 ; foo2 -> func1 ();

foo2 -> func2 ();

foo3 = std :: shared\_ ptr < Foo >( new Foo Kid ()); foo3 -> func1 ();

foo3 -> func2 ();

Foo Kid foo4 ;

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

Foo foo5 ( foo4 );

foo4 . func1 () foo4 . func2 ()

foo5 . func1 () foo5 . func2 () return 0;

}

18

19

20

21

22

23

24

25

What would the output of the code be when main is run? For each line that is printed, please specify which object and method is causing the line to be printed.