

MF 703
Programming for Math Finance
Practice Midterm Exam

Professor Christopher Kelliher

November 5, 2020

Name: _____

BU ID: _____

Signature: _____

Problem 1 (10 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)

Table names are in bold, columns are listed in each line and data types are in parentheses.

1. (3 Points) Write a query using only the **instrument_prices** table that selects a time series of historical closing prices for a given ticker. Make sure the results are sorted from oldest date to newest.
2. (3 points) Write a query that selects the closing price and shares outstanding from the **instrument_info** and **instrument_prices** tables for a given ticker and quote_date. Only return the data if the instrument has a price in the **instrument_prices** table.
3. (3 points) Write a query that computes the average daily return for each sector for a given date.
4. (1 point) You are asked to remove one redundant column from the **instrument_prices** table. Which column would you remove and why?

Problem 2 (10 points) Python Basic Concepts

1. (2.5 points) Consider the following code:

```
1 str1 = "advanced"
2 print(str1)
3
4 str2 = "programming"
5 print(str2)
6
7 str = str1 + str2
8 print(str)
9
10 test = "gram" in str
11 print(test)
12 print(str[3])
13
14 str[1] = "p"
15 print(str)
```

What will be the result of the code? What will be printed to the console?

2. (2.5 points) Consider the following code that implements the Black-Scholes formula for a European call:

```
1 def callPx(s_0 , k, r, sigma , tau ):
2     sigmaRtT = ( sigma * math . sqrt ( tau ))
3     rSigTerm = (r + sigma * sigma / 2.0) * tau
4     d1 = ( math.log (s_0 / k) + rSigTerm ) / sigmaRtT
5     d2 = d1 - sigmaRtT
6     term1 = s_0 * norm.cdf(d1)
7     term2 = k * math.exp (-r * tau ) * norm.cdf (d2)
8     return term1 - term2
```

What happens when a negative value is passed for `s_0`? How about a negative `sigma`? Add the proper exception handling to this function to handle these parameter values.

3. (2.5 points) You are given two DataFrames which contain historical price data for two different ETF's. Unfortunately, the dates in the two data frames don't match exactly. Write the Python code necessary to merge these two data frames, returning only dates where there are prices for both ETF's.
4. (2.5 point) Suppose `mat1` and `mat2` are two Python numpy matrix objects, and `arr1` and `arr2` are numpy array objects.

If we same data is stored in `mat1` and `arr1`, and `mat2` and `arr2`, respectively, what will be the difference, if any, in behavior between multiplying **`mat1 * mat2`** and multiplying **`arr1 * arr2`**?

Problem 3 (20 points) Object Oriented Programming in Python

1. (10 points) Consider the following Python code:

```
1 class BasePosition:
2     def __init__(self, shrs, long):
3         self.shares = shrs
4         self.isLong = long
5         print("calling BasePosition class constructor")
6
7     def __del__(self):
8         print("calling BasePosition class destructor")
9
10    def printPos(self):
11        print("calling printPos from Base")
12        print(self.shares)
13        print(self.isLong)
14
15 class ChildPosition(BasePosition):
16     def __init__(self, shrs, long, childShrs):
17         self.childShares = childShrs
18         BasePosition.__init__(self, shrs, long)
19         print("calling ChildPosition class constructor")
20
21     def __del__(self):
22         print("calling ChildPosition class destructor")
23
24     def printPos(self):
25         print("calling printPos from Child")
26         print(self.shares)
27         print(self.isLong)
28         print(self.childShares)
29
30 basePos1 = BasePosition(100, 1)
31 basePos2 = BasePosition(75, 0)
32
33 childPos1 = ChildPosition(25, 0, 5)
34 childPos2 = ChildPosition(150, 1, 0)
```

What will be the output of this program?

2. (5 points) What are the `__del__` functions and at what point will they be called?
3. (5 points) What will be the result of the following line of code?

```
1 basePos3 = basePos1 + basePos2
```

What, if any changes would need to be made to the `BasePosition` class for this to work?

Problem 4 (10 points) C++ Basic Concepts

Consider the following piece of C++ code:

```
1 class Foo{
2     Foo(int bar_){
3         bar = bar_;
4     }
5
6     int bar;
7 };
8
9 int main(int argc, const char * argv[]) {
10     Foo f;
11     f.bar = 5;
12     std::cout << f.bar << std::endl;
13
14     return 0;
15 }
```

1. (4 points) Will this code compile? If not, fix any errors so that it will compile. If it will compile, what will the output be?
2. (4 points) Consider the following set of functions added to the Foo class:

```
1 int calcFooBar1(int val, int mult){
2     bar = val * mult;
3     val = bar;
4     return bar;
5 }
6
7 int calcFooBar2(int& val, int& mult){
8     bar = val * mult;
9     val = bar;
10    return bar;
11 }
12
13 int calcFooBar3(int* val, int* mult){
14     bar = (*val) * (*mult);
15     val = &bar;
16     return bar;
17 }
```

(1 point) Which calcFooBar function takes a pointer?

(3 points) What will be the output of the following code?

```
1 Foo f;
2 int val = 1.0;
3 int mult = 5.0;
4
5 int ret = f.calcFooBar1(val, mult);
6
7 std::cout << val << std::endl;
8 std::cout << mult << std::endl;
9 std::cout << ret << std::endl;
10
11 val = 1.0;
12 mult = 5.0;
13
14 ret = f.calcFooBar2(val, mult);
15
16 std::cout << val << std::endl;
17 std::cout << mult << std::endl;
18 std::cout << ret << std::endl;
19
20 val = 1.0;
21 mult = 5.0;
22
23 ret = f.calcFooBar3(&val, &mult);
24
25 std::cout << val << std::endl;
26 std::cout << mult << std::endl;
27 std::cout << ret << std::endl;
```

3. (2 point) Consider the function calcFooBar4:

```
1 int calcFooBar4(const int& val, const int& mult){
2     bar = val * mult;
3     val = bar;
4     return bar;
5 }
```

What will be the result of adding this function to the Foo class and calling it from the main function?

Problem 5 (20 points) Object Oriented Programming in C++

Consider the following C++ code:

```
1  class Foo{
2      public:
3          Foo(){
4              std::cout << "calling Foo constructor" << std::endl;
5          }
6
7          virtual ~Foo(){
8              std::cout << "calling Foo destructor" << std::endl;
9          }
10
11         int bar;
12     };
13
14     class FooKid : public Foo {
15     public:
16         FooKid(){
17             std::cout << "calling FooKid constructor" << std::endl;
18         }
19
20         virtual ~FooKid(){
21             std::cout << "calling FooKid destructor" << std::endl;
22         }
23     };
24
25     int main(int argc, const char * argv[]) {
26         Foo* f(new Foo());
27         Foo f2(*f);
28
29         FooKid* fk;
30         std::cout << (f == &f2) << std::endl;
31
32         delete f;
33
34         std::shared_ptr<Foo> f3;
35         fk = new FooKid();
36         f3 = std::shared_ptr<Foo>(new Foo());
37
38         return 0;
39     }
```

1. (10 points) What will be the output of the above code?
2. (2.5 points) Consider adding a function `func()` to the definition of `Foo`. What would be the significance of making the function virtual? In what cases would this be useful?
3. (2.5 points) When specifying that `FooKid` inherits from `Foo`, what is the significance of the `public` keyword? What would happen if instead this line read:

```
1 class FooKid : Foo
```

4. (2.5 points) Are there any dangling pointers created in the above code? If so, what code would you change to fix this?
5. (2.5 points) Suppose the following function definition was added to the definition of `Foo`:

```
1 virtual void printFunc() = 0;
```

What type of function is this? What impact would adding this have?

Problem 6 (20 points) Simulation Algorithm

Consider the Black-Scholes SDE:

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

You are asked to use this SDE to write a simulation algorithm to price an American upside one touch option. Recall that the payoff of an American one touch is defined as:

$$c_0 = \tilde{\mathbb{E}} \left[e^{-rT} 1_{M_T > K} \right]$$

where M_T is the maximum value of the asset over the period.

1. (2.5 points) List the set of parameters that you will have in your algorithm and describe their meaning.
2. (7 Points) Write a piece of pseudo-code that you can use to simulate from the given stochastic process.
3. (7 Points) Write a piece of pseudo-code that will define the payoff function for your exotic option.
4. (3.5 points) Describe three unit tests that you would create to ensure that your model prices are correct.

Problem 7 (10 points) Algorithms & Finance Applications

1. (2.5 points) What is the singleton pattern? When should it be used?
2. (2.5 points) Using pseudocode, write a brief sketch of the code needed to implement the singleton pattern.
3. (2.5 points) Estimate the sign and rough magnitude of the correlation between the S&P 500 and its implied volatility.
4. (2.5 points) Suppose we buy a straddle on the S&P and delta-hedge it continuously. What factors will determine our profit or loss on the trade?

