CS 5523: Bonus Programming Projects

You may choose to work on either of the following two programming projects. Submission of successful code of either project will get 5 bonus points. You may choose both, but only get maximum 5 bonus points instead of 10.

Project 1 (+5): Math Client/Server – Middleware-RPC

Objectives:

- Enhance the understanding of client-server architecture of distributed systems
- Practice the usage of RPC

You will design and implement a **Math server** that provides the following **four** remote procedures:

- *magicAdd*( ): takes 2 **double** parameters and returns the **difference** between the 2 values; that is it actually does subtraction operation;
- *magicSubtract*( ): takes 2 **double** parameters and returns the sum of the 2 values;
- *magicFindMin*( ): takes 3 **int** values as parameters and returns the **largest** value; and
- *magicFindMax*( ): takes 3 **int** values as parameters and returns the **smallest** values.

Moreover, the **Math server** should keep some counters to record the number of different operations it has performed, and the corresponding methods to retrieve these numbers.

On the client side, a client will generate 1000 RPC requests, where each request randomly chooses one of the 4 operations as well as the corresponding required parameters. At the end, the client should retrieve the number of operations performed by the server.

Your program should support running more than one (at least two) client concurrently.
The server and client programs should be able to run on separate machines. You can program with any language.

Report: Write a project report that should include the following:

- The status and design of your project.
- Figures of test results that demonstrate successful design

For program codes and electronic copy of your report: zip them to a single file.

Project 2 (+5) Math Client/Server – Middleware-RMI

- **Objectives**
  - Practice the usage of middleware, such as Java RMI and CORBA

- **Project Description**

You will design and implement a **Math server,** that create a **remote Math object** to provide the following **four** methods:

- *magicAdd*( ): takes 2 **double** parameters and returns the **difference** between the 2 values; that is it actually does subtraction operation;
- *magicSubtract*( ): takes 2 **double** parameters and returns the sum of the 2 values;
- *magicFindMin*( ): takes 3 **int** values as parameters and returns the **largest** value; and
- *magicFindMax*( ): takes 3 **int** values as parameters and returns the **smallest** values.

Moreover, the **Math object** should keep some counters to record the number of different operations it has performed, and the corresponding methods to retrieve these numbers.

On the client side, a client will generate 1000 RMI requests, where each request randomly chooses one of the 4 operations as well as the corresponding required parameters. At the end, the client should retrieve the number of operations performed by the server.

Your program should support running more than one (at least two) client concurrently.
The server and client programs should be able to run on separate machines.

Report: Write a project report that should include the following:.

- The status and design of your project.
- Figures of test results that demonstrate successful design

For program codes and electronic copy of your report: zip them to a single file.

Requirements:
You must submit your work using Blackboard Learn and respect the following rules:

1) **All assignments must be named as Firstname_id.**
2) **If the assignment includes programming, the source file should be submitted.**
3) **Assignment folder must include source file(s).**
4) **Draw a figure on the paper and take a photo of it to include in the homework if you don't want to draw the electrical version.**