Work Report

Main Points:

1- **Reading file from user.**
2- **Sending the file content to the server.**
3- **Calling the wrap up class and give it the vector containing the instructions.**
4- Allocating variables according to the instructions included.
5- Specifying the operation needed and executing it.
6- **Return results to the client.**

Reading from user

At the beginning the user insert a text file into the program with the instruction included to be executed. At this point the program takes each line of the instruction and separate it into two parts using Regular Expression. The first part is the code of the instruction such ADD or DIV, and the second part is the variables part such as $myInt,100,50,200. These two parts will be stored into vectors and the odd indices will be the code instruction and the even indices will be the variables.

Sending the file to the server

After reading the file content and putting it into a buffer, the buffer is sent to the server and is read on the server side. Also the content of the file is taken from the buffer and is put inside a vector of strings to be sent to the wrap up class.

Then, the file content is sent to the wrap up class as a parameter of its constructor when creating an object of the class. The vector content is stored in the vector called lines in the wrap up class to be processed and to carry out the instructions specified in each line.

Allocating variables

At this stage there are two main classes made to complete the process. The Variable class which includes the main types of the data given to the process such Numeric or Real, and the other main class is Operation class which includes all the operations will executed according the code instruction given in the text file inserted. There are two scenarios in this stage, the first is the Code instruction from the vector is Var, then it calls a function that takes the other parameter given and insert it into map. The other scenario is that code instruction is operation needed to be executed such DIV, then it goes to the next stage of the process.

Specifying the operation

At this point it is know that we got a code instruction with operation word like Div or Add, so it goes to the operation class and calls the specific function needed then take the parameters from the second part of map which include Both Code instruction and Variables and execute the operation.

Return the results to the user

This stage doesn't require to print all the results comes from each line of instruction, since it only requires to print the results come from OUT operation into the output file and the errors to the error file.

Command used to build the program:

Server Side:

```
        g++ -std=c++11 charvar.cpp Connection.cpp GarbageCollector.cpp
    numericvar.cpp parse.cpp realvar.cpp server.cpp stringvar.cpp
    TCPServerSocket.cpp TCPSocket.cpp Thread.cpp var.cpp wrap.cpp          Threadexe.cpp
Threadmanage.cpp -lpthread -o server && ./server
```

Client Side:

```
        g++ -std=c++11 client.cpp main_client.cpp -o client && ./client
```

Work of each team member:

Ehab EL-Kady: worked on operations class and retrieving the operations from their map, client and server connection, Multi-threading.

Yousef el-masry: worked on operations classes, VAR classes and storing and retrieving the variables, client and server connection, parsing and syntax error.

Fadi Amged: worked on the UML design, Makefile, Command line Execution, two methods in operations.