**Report on MIS Project**

**Noha Magdy-1610126, Mohannad Barakat-1610563, Hossam Elharmiel- 1610148**

*Department of Computer Engineering, Nile University*

***Abstract*— this report shows how MIS project we provide work**

1. PHASE 1

Inside this phase there are three phases which are:

**First:**

This phase focused mainly on understanding the problem and on doing some research regarding the topic. We reached this conclusion:

1. We need to create 4 major classes: variables class, operations class, jump class, errors class. Beside the main MIS class that includes all the work.
2. Each class will inherit other different classes to create different variables and operations.
3. The MIS main class contain different function to:

* Create different files
* Read and write on these files
* Split the words on files
* Parse the files
* Check if the line is for variable or operation
* Perform jump or assign
* Throw errors when needed

1. The Errors classes maintain all different kinds of errors for different variables and operations.

**Second:**

This phase is for designing the UML on visual paradigm program to help on implementing the codes. 3 types of diagrams are used

* Use case diagram
* Sequential diagram
* Class diagram

All the diagrams are attached with the project files

**Third:**

This phase is for implementing the codes on the files attached using c++ 17 programing language and object oriented programing concepts.

**To Run the project:**

g++ -std=c++17 ADD.cpp DoubleJump.cpp JMPGTEError.cpp JMPLTE.cpp JMPNZ.cpp JMPZError.cpp MisError.cpp NotDeclared.cpp REAL.cpp SUB.cpp VAR.cpp CHAR.cpp JMPGT.cpp JMPGTError.cpp JMPLTEError.cpp JMPNZError.cpp Jump.cpp MUL.cpp NUMERIC.cpp SingleJump.cpp SyntaxError.cpp ZeroDiv.cpp DIV.cpp JMPGTE.cpp JMPLT.cpp JMPLTError.cpp JMPZ.cpp MIS.cpp nonNumeric.cpp OPERATION.cpp STRING.cpp undefined.cpp ASSIGN.cpp -o try

**Then run** :

./try

1. PHASE 2

The second phase is the networking phase. We implemented both UDP and TCP protocols. The communication was divided into the client part and the server part.

**The client:**

The client is responsible for preparing the MIS file and dividing it into packets then send the buffer through the client socket.

**The server:**

The server is responsible for putting the buffer into a vector, parse it, execute it through MIS, and send the results and errors through the server socket.

III. PHASE 3

This phase is for the multi-threading part. Multiple clients could send different MIS programs to one server. The server is responsible for creating different threads for each client.

To run the whole project:

server:

1- make

2- misvm

client:

1- make

2- client\_misvm <server\_ip> <port\_number> <file\_name\_without\_extention>