CPU Simulator (Von Neumann Architecture)

By: Shudong Lai, Yifan Chen, Hitesh Agarwal and Rahul Sangewar

Introduction:

This is a Java application designed to store the users input from hexadecimal values to binary

and then display it on the interface. It needs a Java version1.8 environment or higher. It shows the values from a text file in a “*.txt” format only that are represented in hexadecimal values, storing it into the system memory on the left pane of the GUI. The available functions are LDR and STR. There’s an input text box that you can input instructions. No other instructions are implemented yet.

Design:

Our project has 4 working classes.

GUI class: it is the frontend interface the program runs in it. It uses all other classes.

Register class: Register class is made up of address and value. All registers can be implemented by this class.

Memory class: It is mainly made up of a 2d string array that is String[address][value]. Memory ‘s values are initiated as ‘0000,0000,0000,0000’.

Instruction class: It’s made up of memory and registers. It is the backend of the program.

Usage:

Load button: The user can see how the program works by clicking the “Load” button on the GUI’s interface, picking their chosen .txt file (structured txt file only). The values are already loaded to the system memory when they “load” their input file during the operation of the load button and is shown in the output window.

Run button: When you enter the instructions in the input box, you can hit run button and the instruction can be processed.

Reset button: All memory cleaned and reset.

Exit button: Immediately exit the program.

Output textbox: Show used memory addresses and values.