CSCI6461 Team Project Phase 1, 2 and 3

User Manual

Team Details -

`Team Number - 7`

Name	GWID	Email
Vaibhav Vemula	G37091446	vaibhav.vemula@gwu.edu
Rajith Ravikumar	G45692898	rajith.ravikumar@gwu.edu
Gayathri Kalthi Reddy	G33650136	gayathri.kalthireddy@gwu.edu
Srinith Rao Bichinepally	G37406626	srinithrao.bichinepally@gwu.edu

---

### 1. Installation

Requirements - `JDK-19+`

- Extract the zip file and open the directory. OR
- Git clone the repo using the following command

```
git clone https://github.com/vaibhav-vemula/CSA_Simulator
```

2. How to run?

Look for a **jar** file called `CSA_Simulator.jar`. It will run the simulator. OR Look for `Main.java` file and execute the file using following command in same directory
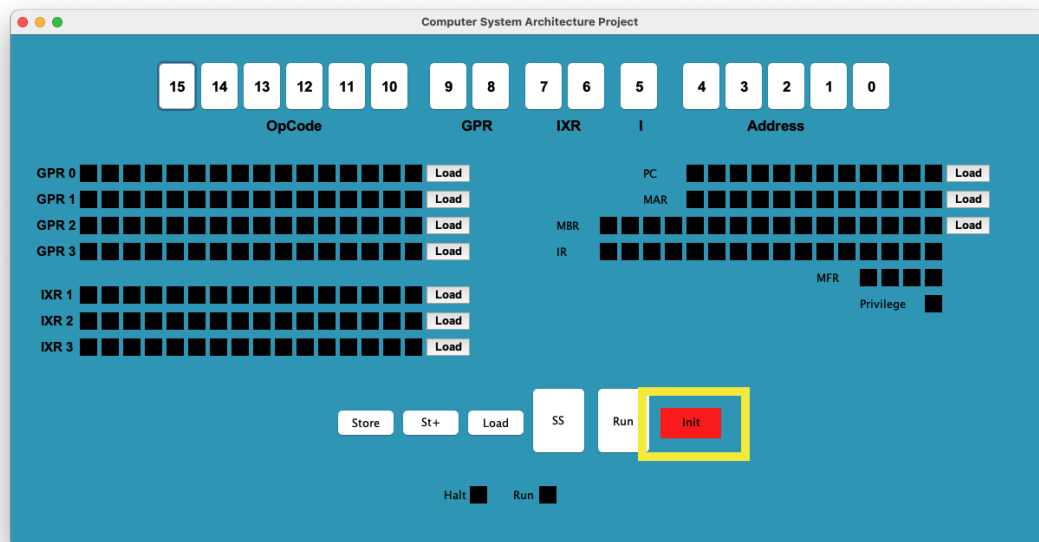
```
javac Main.java && java Main
```
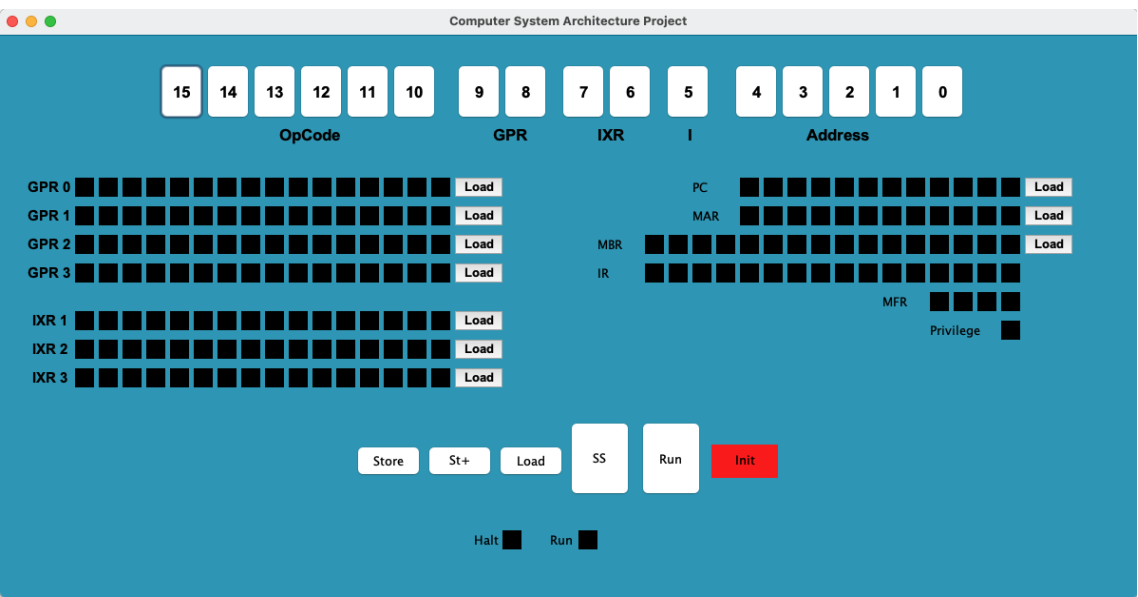
-> PHASE 1

3. Running the simulator

To run the program click on the init button (highlighted in yellow in fig. below) that would bring the file upload dialogbox.


Navigate the Program folder and choose which program you would like to run. On Accepting the file click ok then you will see in the console the lines of code that has been loaded into the memory.
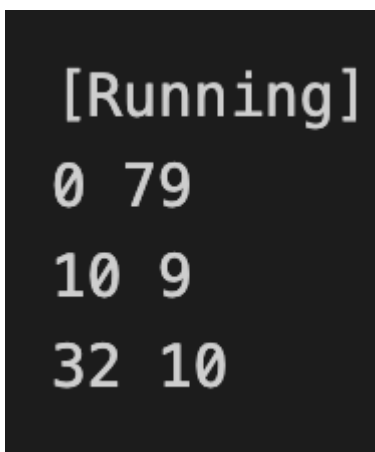

4. Testing the simulator

- Press those switch buttons on top to set the value where entry point of the program is located.

- Load the value in the switch to the Program Counter (PC) by pressing the LD button adjacent to it.
- Once the Value is loaded it would turn on the simulated LEDs.
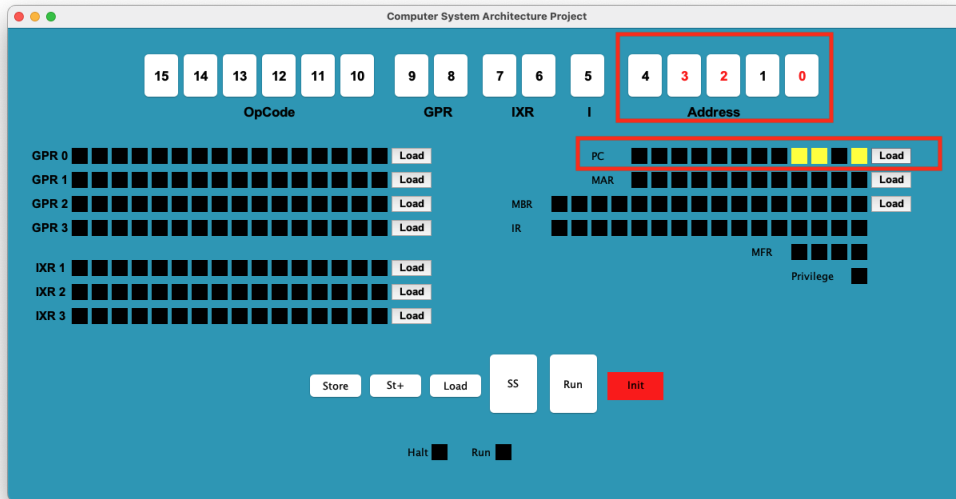
Now there are two options to choose to test the simulator

- You can test the program step by step using the `SS button`. It will traverse through the program normally as the Program Counter is incremented.
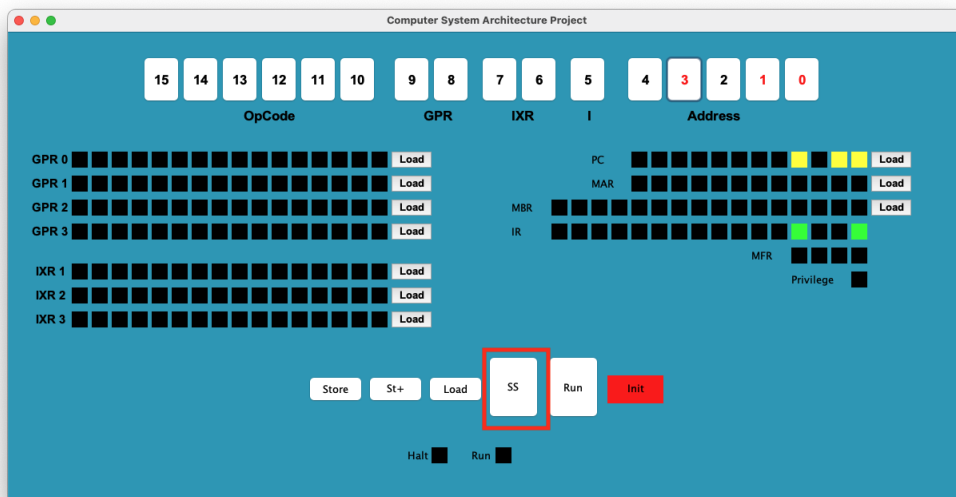

- OR

- You can run the program directly by using the `run` Button. As soon as the Run button is pressed, the PC will increment or jump to other instructions on its own and the Run indicator will turn GREEN. Once the program comes to a halt, the halt LED indicator will turn RED and the Run indicator will turn BLACK.

After this no more programs can be run unless it is reset. The simulator will
be in halt state till it is reset.
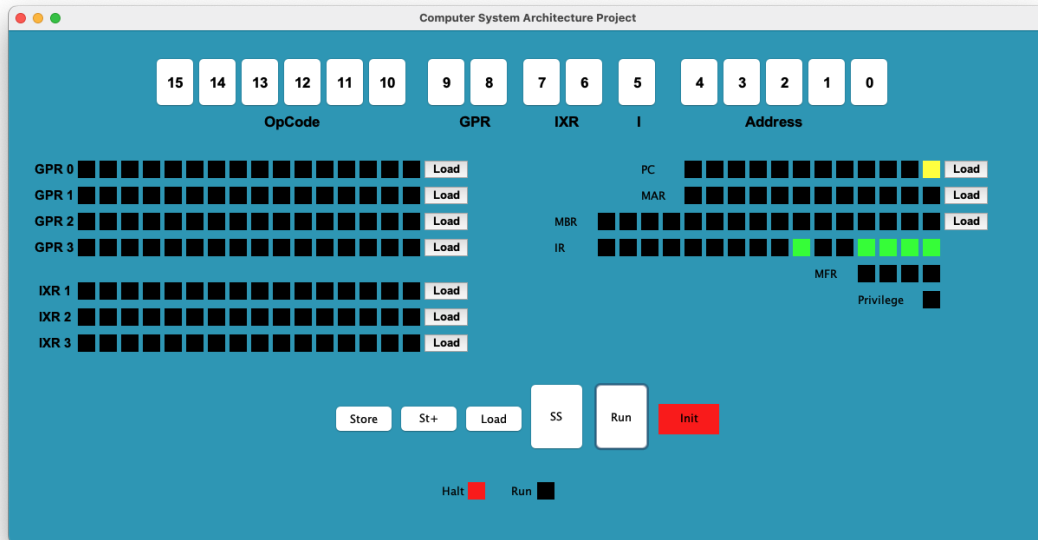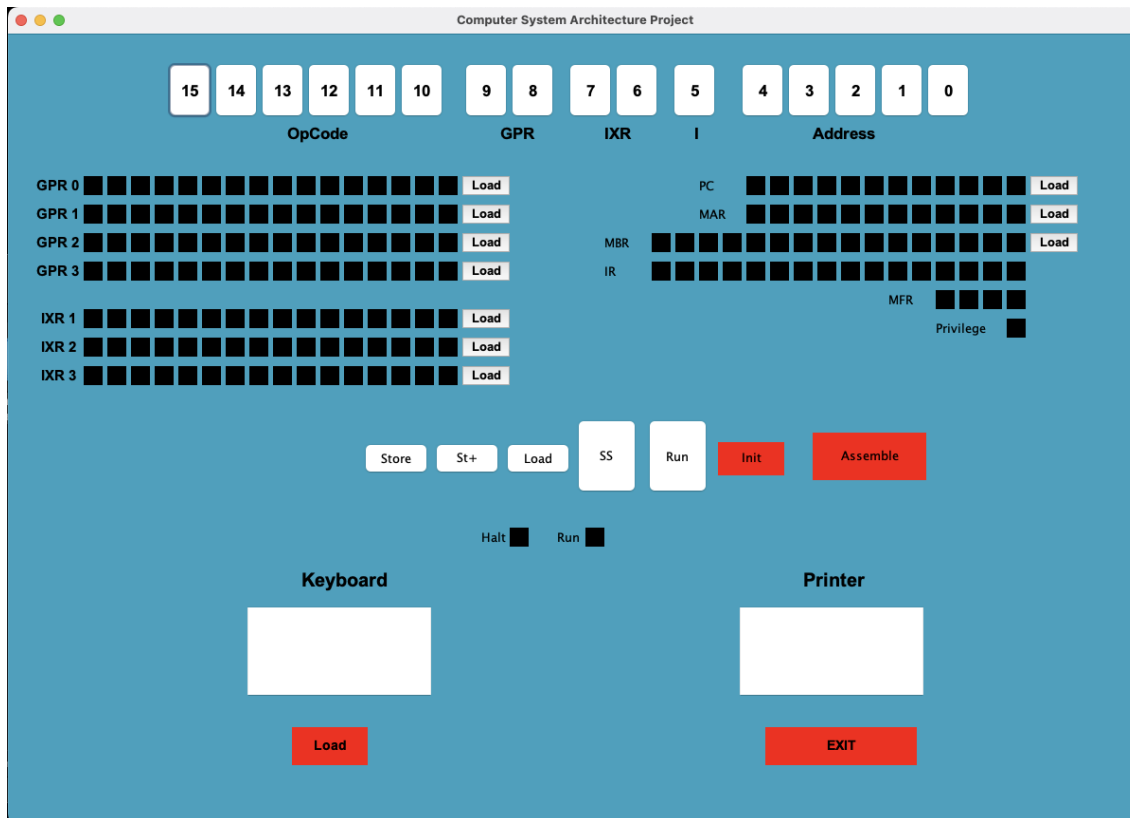

PHASE 2

5. Running Assembler code
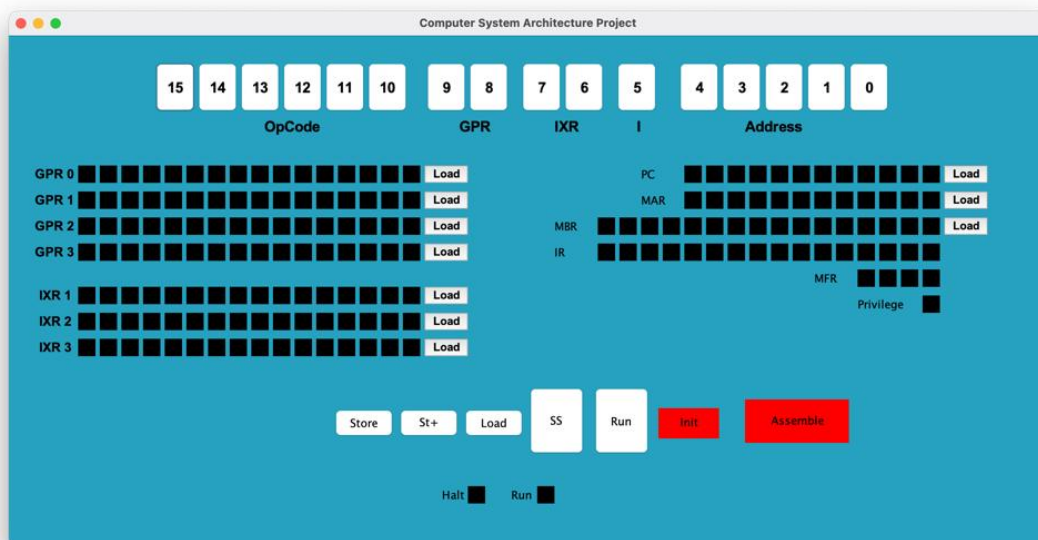
- Follow steps 1-3
- Click on `Assemble` button which will pop up a file selector.

- Choose the input file with set of instructions.
- `AssemblerOutput.txt` will be generated in the folder with encoded hex values.

PHASE 3

6. Running Program 1

- Follow steps 1-3
- Click on `init` button to upload file
- Upload `Program1.txt` file which is present in the root of the project directory.
- Next, Type down 20 numbers in the keyboard and press load to store.
- After entering 20 numbers, enter a number to find the closest number in the pop up.
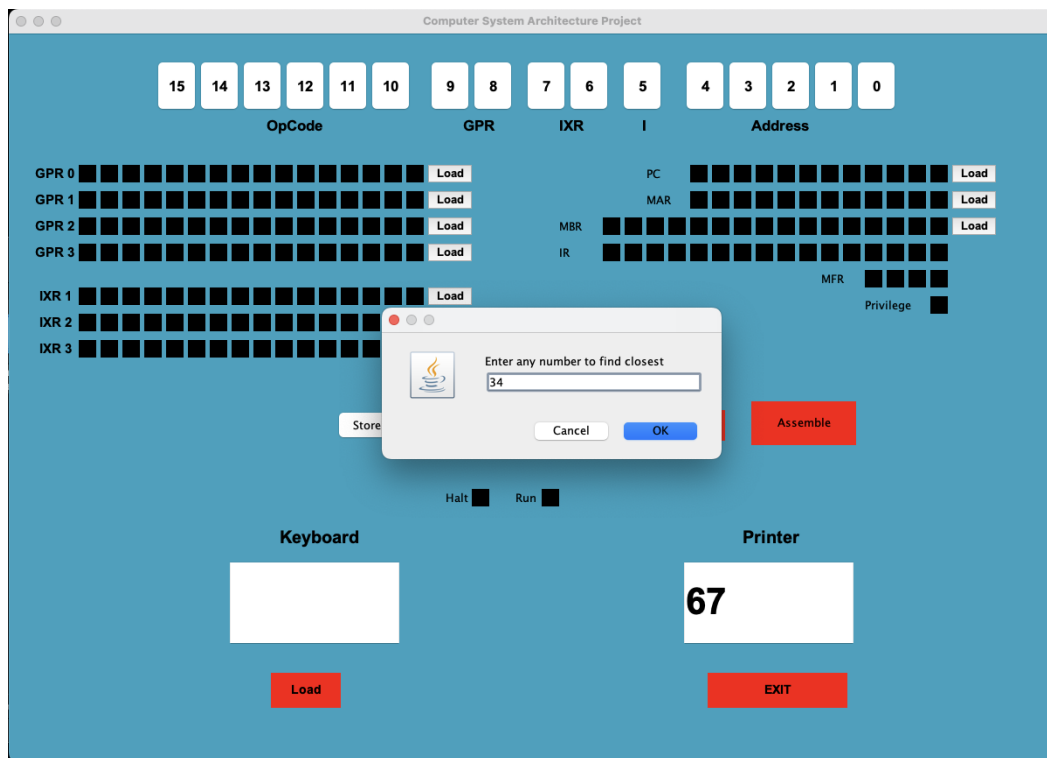
- GUI will be updated to Entered value in the keyboard and the closest number in the printer.