

CSCI 6461 Computer System Architecture

Project Part 1 User Guide

Team 12

Sai Bharath Reddy Lattupalli, Namana Y Tarikere,
Reshma Rajkumar, Vaishnavi Goyal

Preparation Instructions

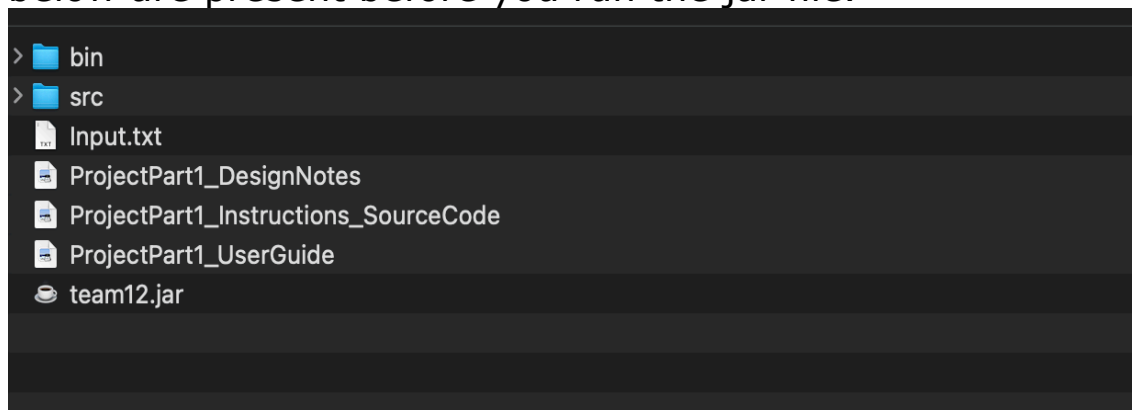
Install Java.

Download the below file from the blackboard.

Filename: Team12_ProjectPart1.zip

Execution Instructions:

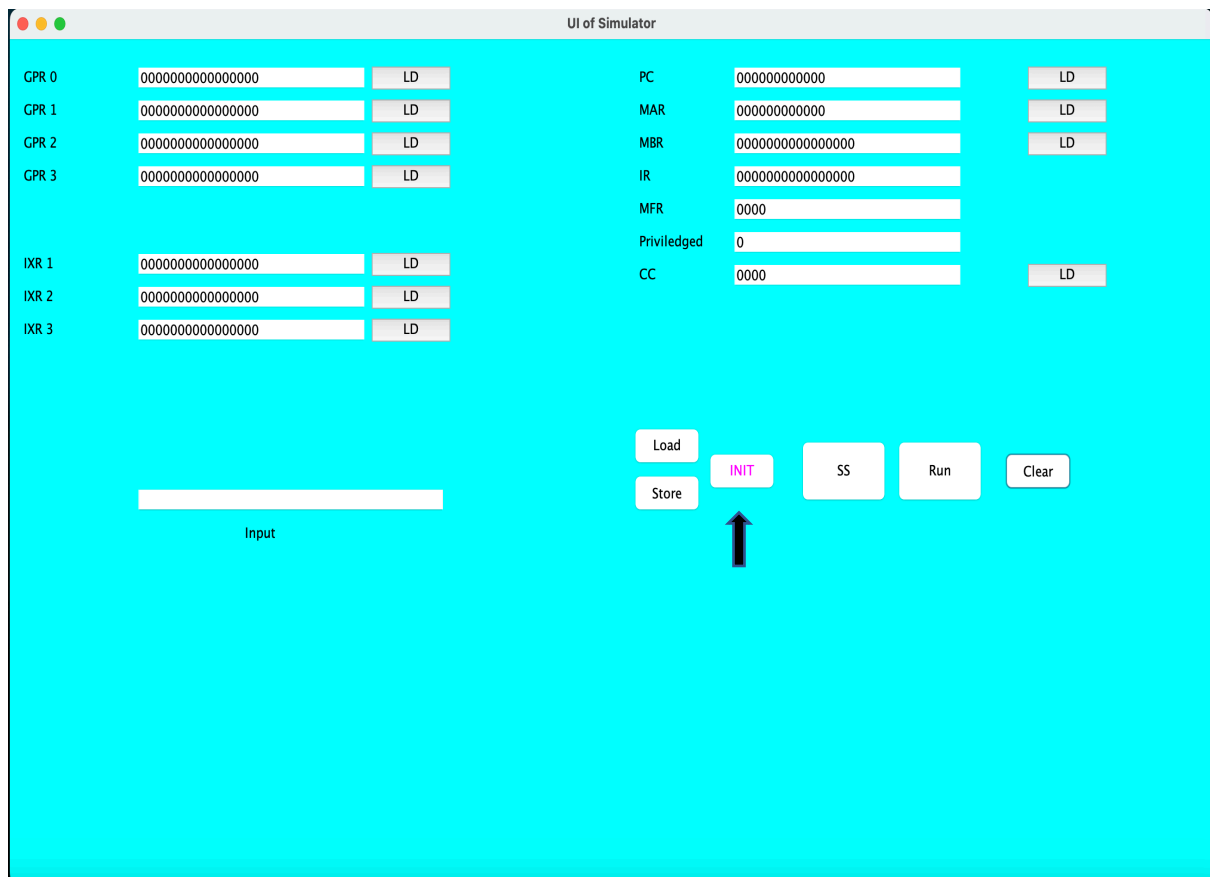
1. Download the file named Team12_ProjectPart1.zip.
2. Extract Zip file and make sure all the files indicated below are present before you run the jar file.



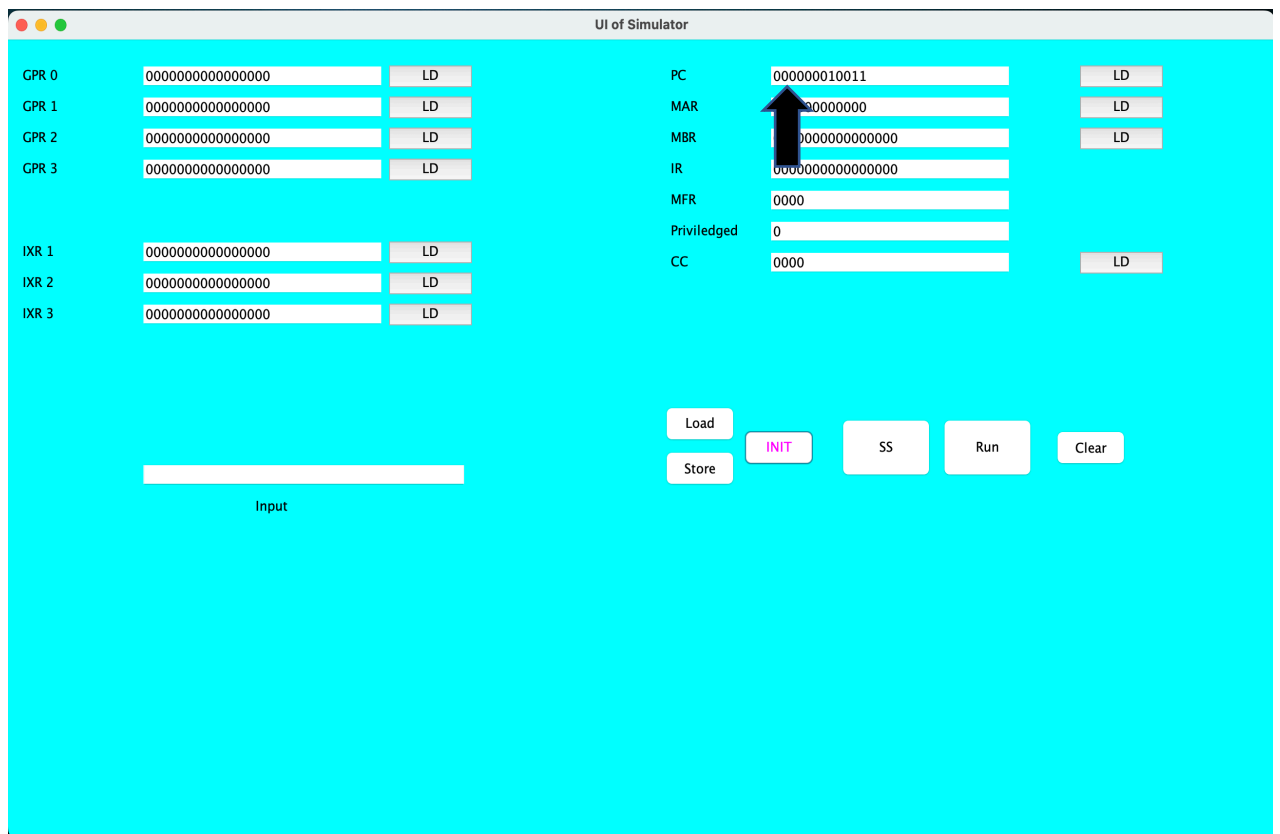
3. Double click on team12.jar to run the jar file/ open terminal and execute "java -jar team12.jar" from the folder where you have these files.

Operating the Simulator

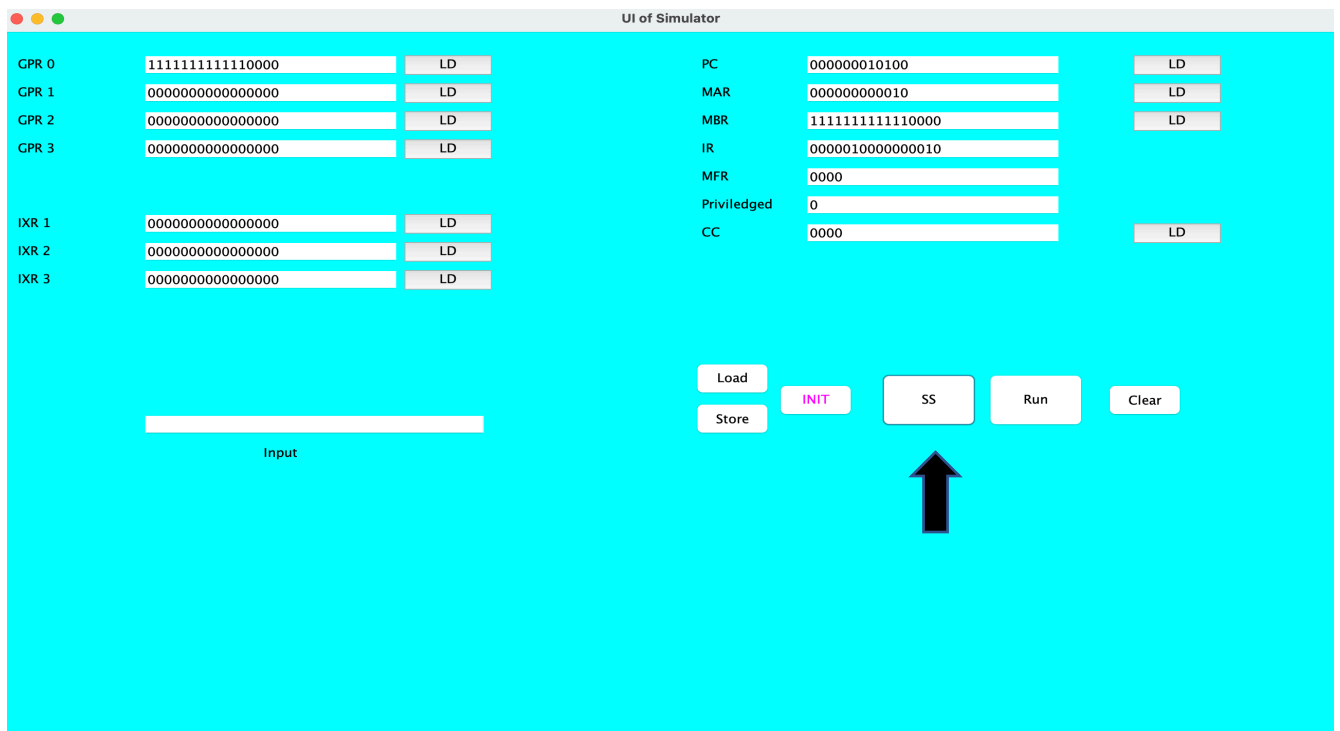
1. Click on INIT button.



The values of default file "input.txt" gets loaded into the memory with the starting address being x0013 i.e., 00000000000010011 in binary form. PC points to this address.



2. Click on “SS” to execute the instructions single step at a time.



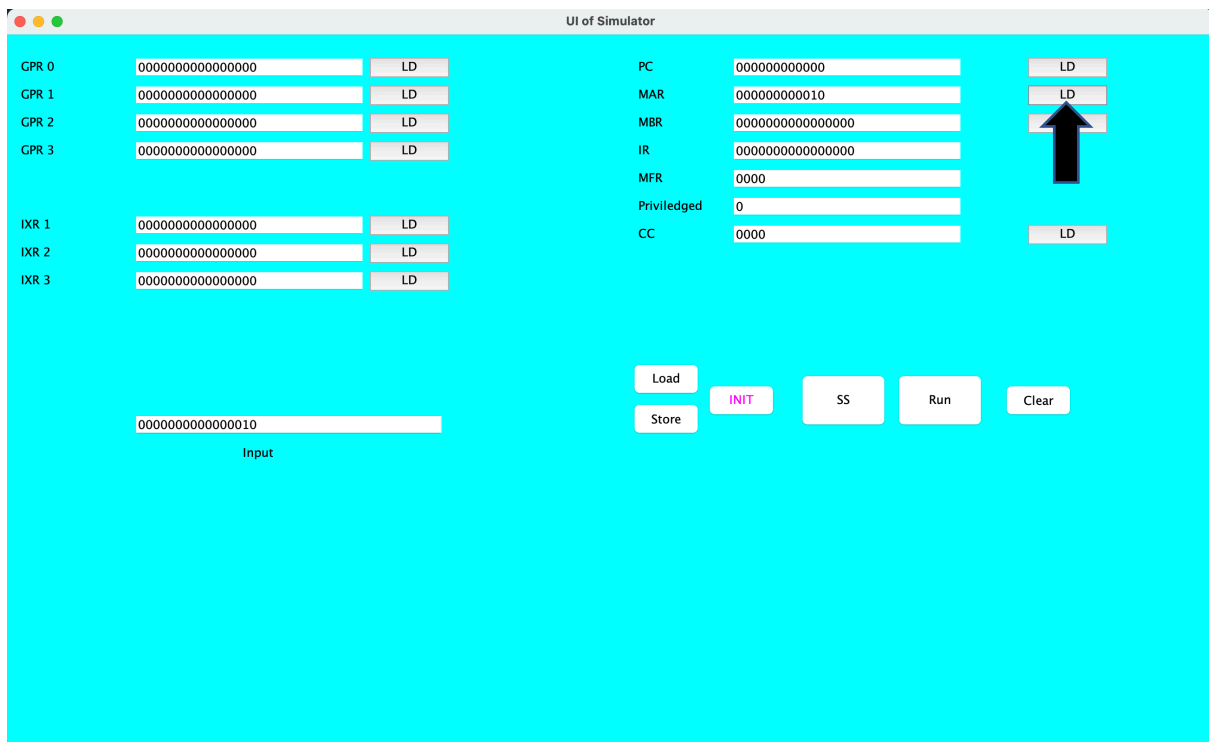
3. After performing “ss” on all instructions that were loaded into memory, click on “Clear” to reset the UI components to their default values.

The screenshot shows a window titled "UI of Simulator" with a light blue background. It contains several sections for registers and control buttons. On the left, there are four General Purpose Registers (GPR 0, GPR 1, GPR 2, GPR 3) and four Instruction Registers (IXR 1, IXR 2, IXR 3). Each register has a text field displaying "0000000000000000" and a button labeled "LD". In the center, there is a large empty text field labeled "Input" below it. On the right, there are several other registers: PC, MAR, MBR, IR, MFR, Privileged, and CC. Each has a text field and a button labeled "LD". The PC, MAR, MBR, and IR fields show "000000000000", while MFR shows "0000", Privileged shows "0", and CC shows "0000". At the bottom right, there is a row of control buttons: "Load", "Store", "INIT" (highlighted in pink), "SS", "Run", and "Clear". A large black arrow points upwards to the "Clear" button.

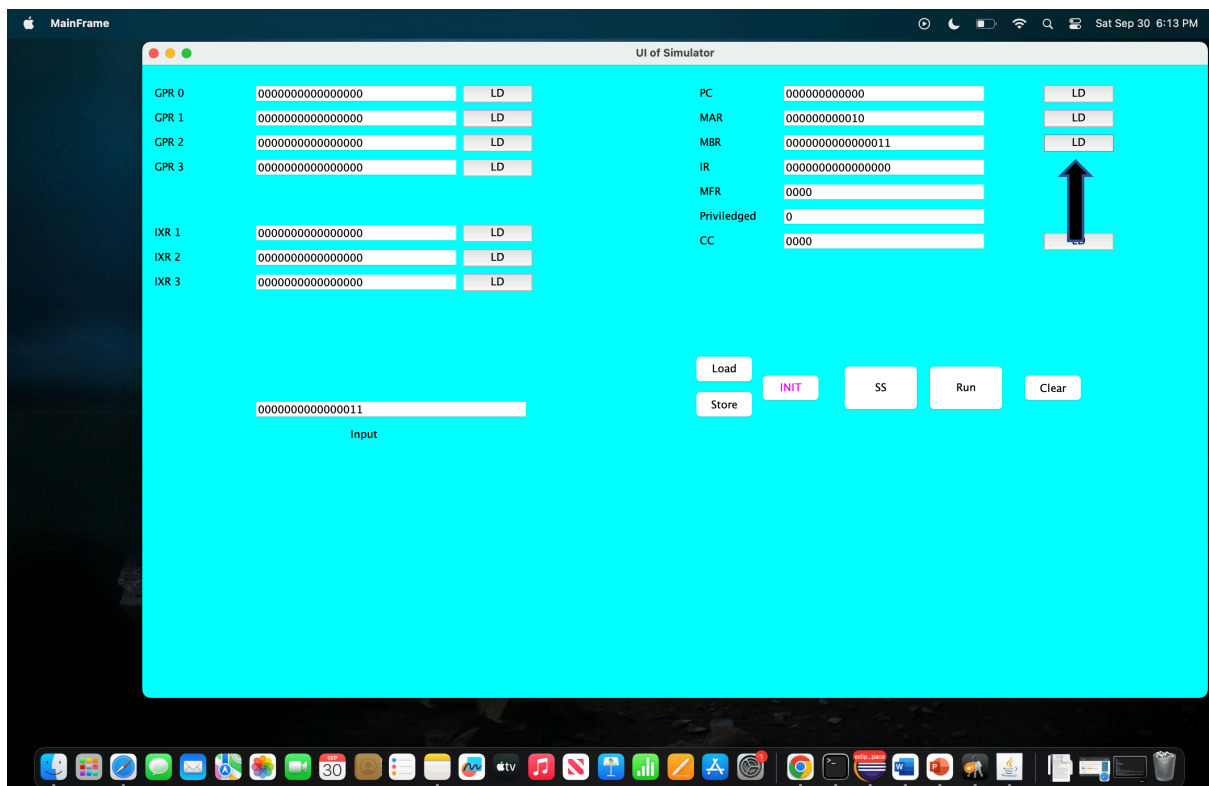
4. Use the textfield named “input” to give manual instructions and then click on any of the “LD” present against the registers to load the value into that specific register.

E.g:

1. To store a value at a particular location, enter the binary address in the input field and then click on “LD” against the MAR register.



2. Then in the input field enter the value to be stored at that location and click on "LD" against MBR.



3. Then click on "Store" button to save this value into the memory.

UI of Simulator

GPR 0	0000000000000000	LD	PC	000000000000	LD
GPR 1	0000000000000000	LD	MAR	000000000010	LD
GPR 2	0000000000000000	LD	MBR	000000000000011	LD
GPR 3	0000000000000000	LD	IR	0000000000000000	
			MFR	0000	
IXR 1	0000000000000000	LD	Priviledged	0	
IXR 2	0000000000000000	LD	CC	0000	LD
IXR 3	0000000000000000	LD			

Input: 0000000000000011

Buttons: Load, Store, INIT, SS, Run, Clear

↑

4. Then click on "Clear" to reset the UI.

UI of Simulator

GPR 0	0000000000000000	LD	PC	000000000000	LD
GPR 1	0000000000000000	LD	MAR	000000000000	LD
GPR 2	0000000000000000	LD	MBR	0000000000000000	LD
GPR 3	0000000000000000	LD	IR	0000000000000000	
			MFR	0000	
IXR 1	0000000000000000	LD	Priviledged	0	
IXR 2	0000000000000000	LD	CC	0000	LD
IXR 3	0000000000000000	LD			

Input:

Buttons: Load, Store, INIT, SS, Run, Clear

↑

5. To load the value that was previously saved, enter the address in the input field and click on "LD" button at MAR.

