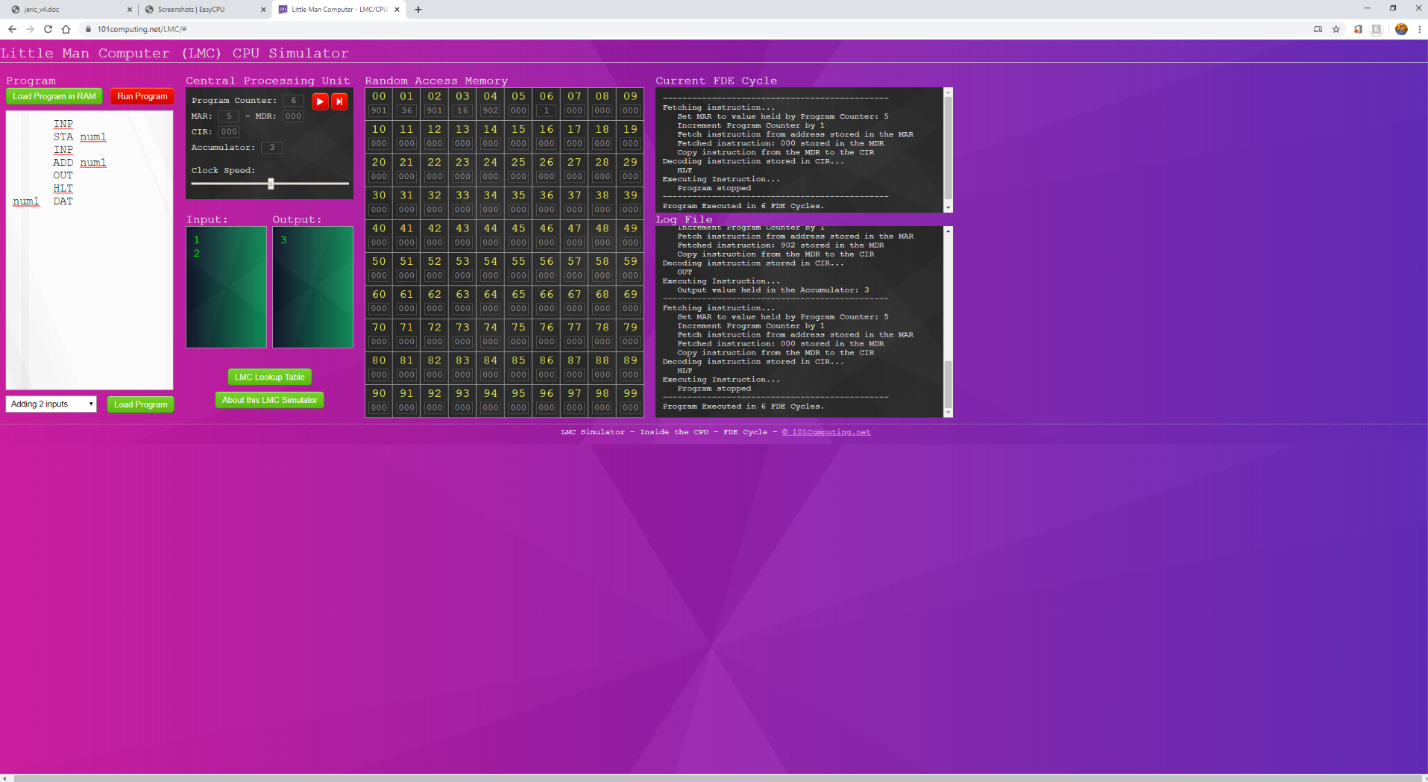
**Step 1:**

Conduct an internet search with you refined key phrase on the subject matter. Report your findings on the space below (e,g, search phrase: “Online Tools for Computer Architecture Performance Simulations”)

**Answer: -**  
Exploring the internet for “Tools for Computer Architecture Performance Simulations”, I have found out the below two tools which are helpful for processor simulations.

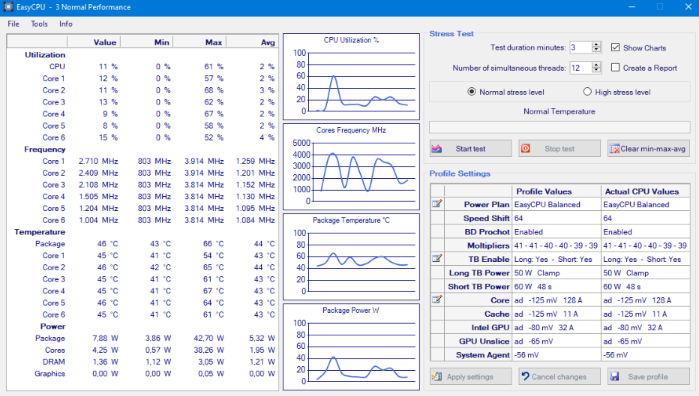
1. “Little Man Computer” is an online tool for processor performance simulations. we will be able to put the inputs and checks the performance of the processor.

When a program is loaded into the RAM and the code is on RUN state, it proceeded from there to Central Processing Unit, having access random access memory, current FDE cycle and LOG file as well.



References: - [https://www.101computing.net/LMC/#](https://www.101computing.net/LMC/)

1. Easy CPU is not an online simulator for checking the performance of a processor simulations. But once downloaded onto the local machine, we will be able to check the real time monitoring of processor sensors like frequencies, temperatures, memory utilizations and so forth.



References: - <https://www.easy-cpu.com/>

**Step 2:**

Explore the contents on this learning resource:

<http://www.ecs.umass.edu/ece/koren/architecture/>

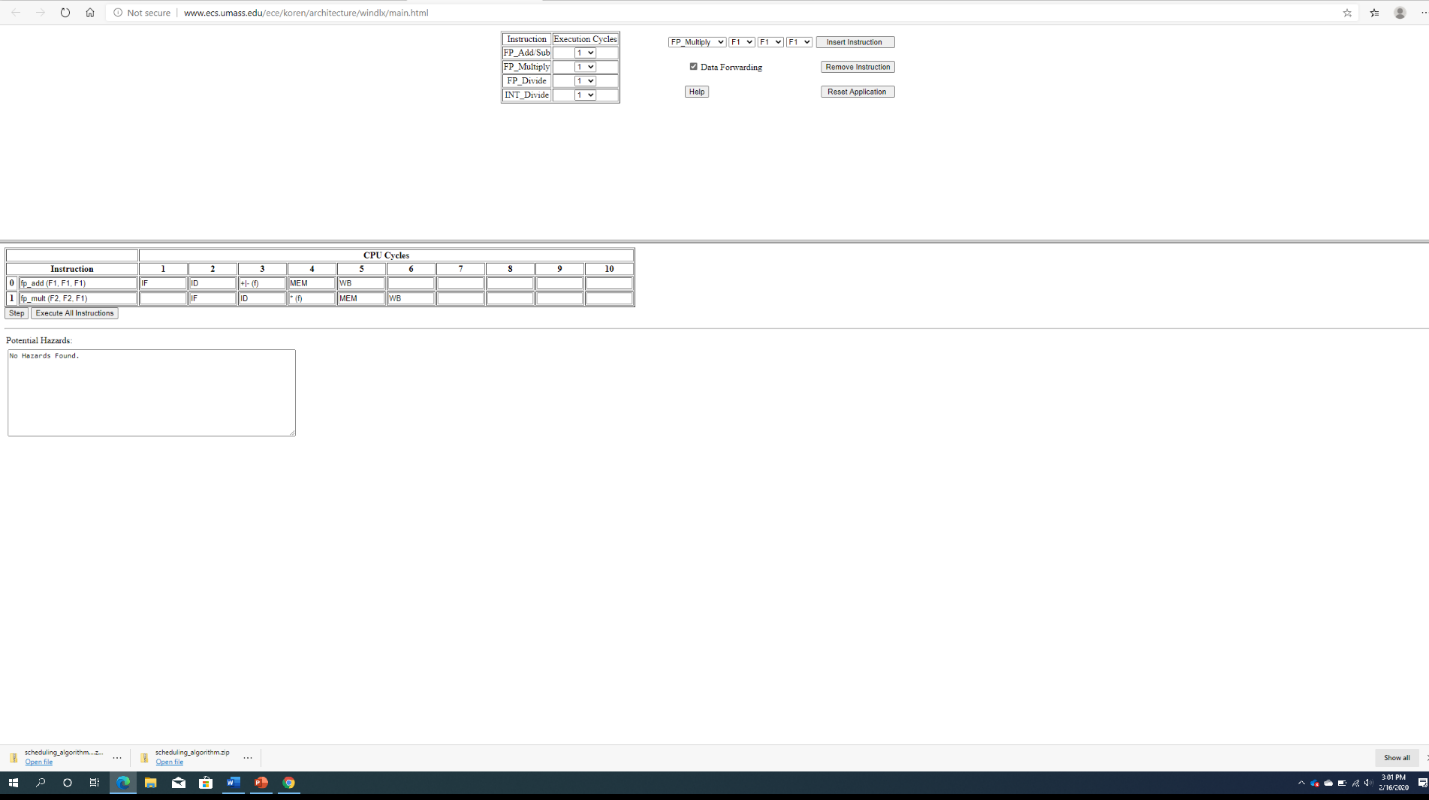
Your explorations and hands-on practices should include at least the following computer architecture topics:

* Cache
* Pipeline
* … <Your topics of interest, e.g. Branch Prediction.> …

**Answer: -**

Capture your practice session results in the form of Demo/Screen\_shots, on the space below. Please be prepared to share your findings with fellow classmates during our live-classroom session.

I have had a chance to explore the Pipelining (W & w/o forwarding) to understand the process of accumulating and executing computer instructions and tasks from the processor via a logical pipeline, allowing to store, prioritize, manage and execute the tasks/instructions in an orderly process.



**Step 3:**

Perstep above, narrow your focus onto 1 specific topic (e.g**. “**Branch Prediction”), and conduct your online search accordingly. Report your findings with hands-on practice demo on the space below.

**Answer: -**

As per the above-mentioned pipelining, we all know that the performance of a processor depends on the following.  
1. Critical Path  
2. Clock Cycle Time  
3. Cycles per Instruction (CPI)

For a single cycle processor, CPI makes the logic and clock simple though the instructions take different time to complete with memory and functional unit are not effectively managed, and the cycle time is more time taking and with lower MIPS and longer clock period, the performance is slower.

For a multi cycle processor, the MIPS is better and have smaller clock period, hence having better performance than a single cycle processor with CPI higher than single cycle processor.

CPU Time = Instruction Count \* CPI \* Clock Cycle Time

**Step 4:**

Explore tools listed in this link:

<http://www.ecs.umass.edu/ece/koren/architecture/links.html>

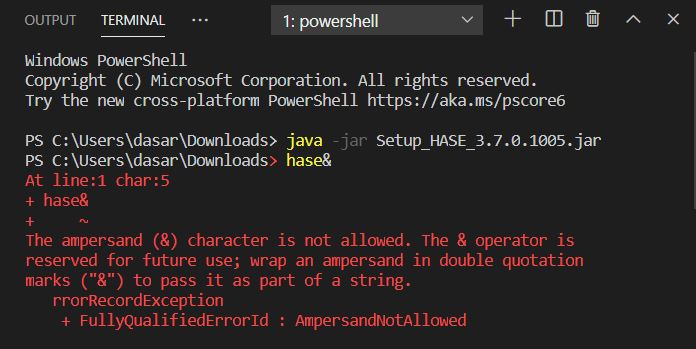
Please report:

* Which tools works, and you find it to be interesting/useful.
* Capture 1 hands-on practice session results on the space below.

**Answer: -**

Reading through the documents of Hase and SimJava from Institute for Computing Systems Architecture on Computer Architecture Simulation and Visualization, being a naïve in computer architecture, it’s something new to me to get to know Hase which is allowing to execute rapid development and detailed analysis in the level of abstraction of data, encompassing both the hardware and software and so forth.

After installing Hase III and trying to run some code, due to systems errors, the assembly code did not work out.



Having said that, I have been exploring on the online tools like “Little Man Computer” and “EasyCPU” as well.

**Step 5:**

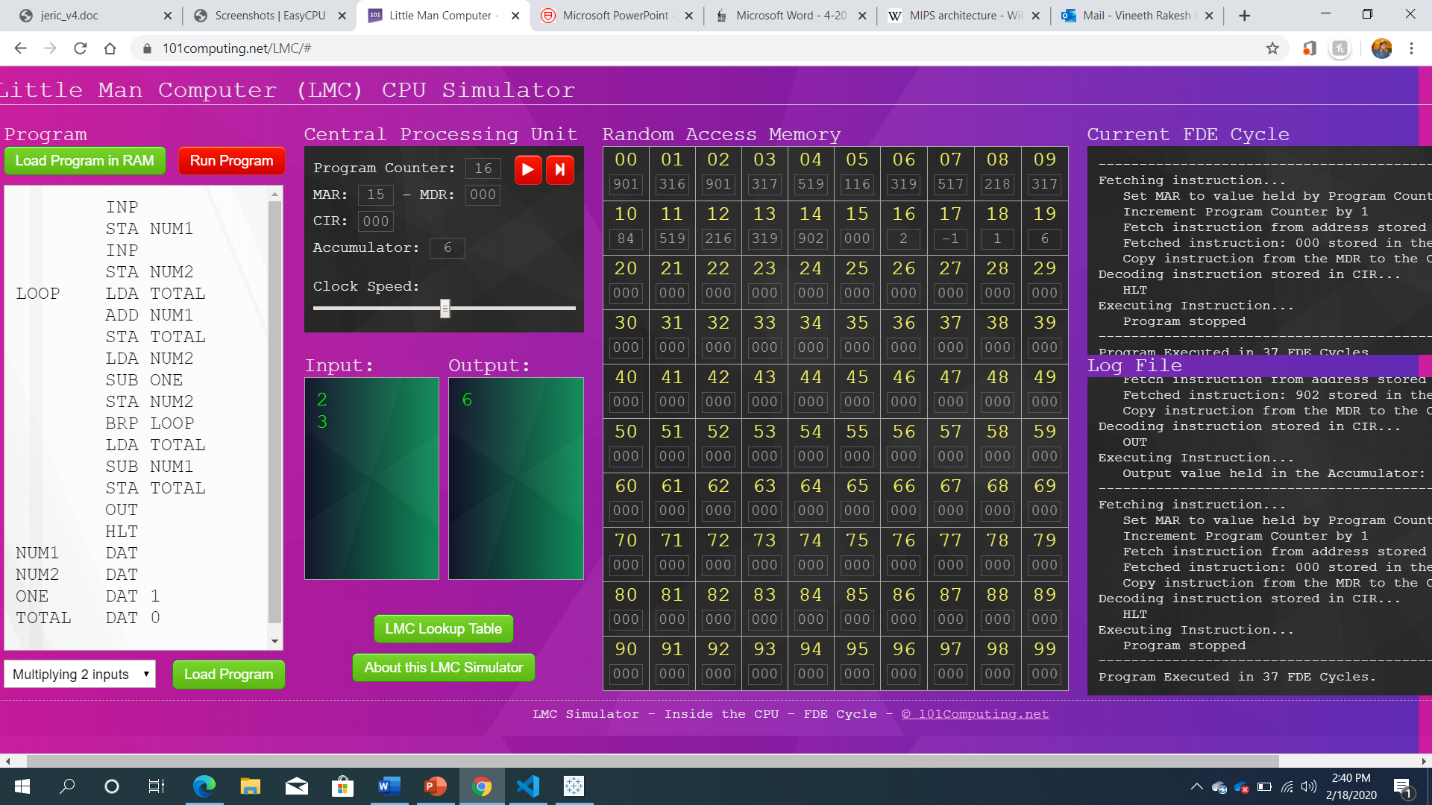
1. Draw your summary/conclusions from this explorations exercise.
2. Your reflections per this link: <http://www.ecs.umass.edu/ece/koren/architecture/contributors.html>
3. Citations/References relevant to your findings.

Please capture your inputs on the space below.

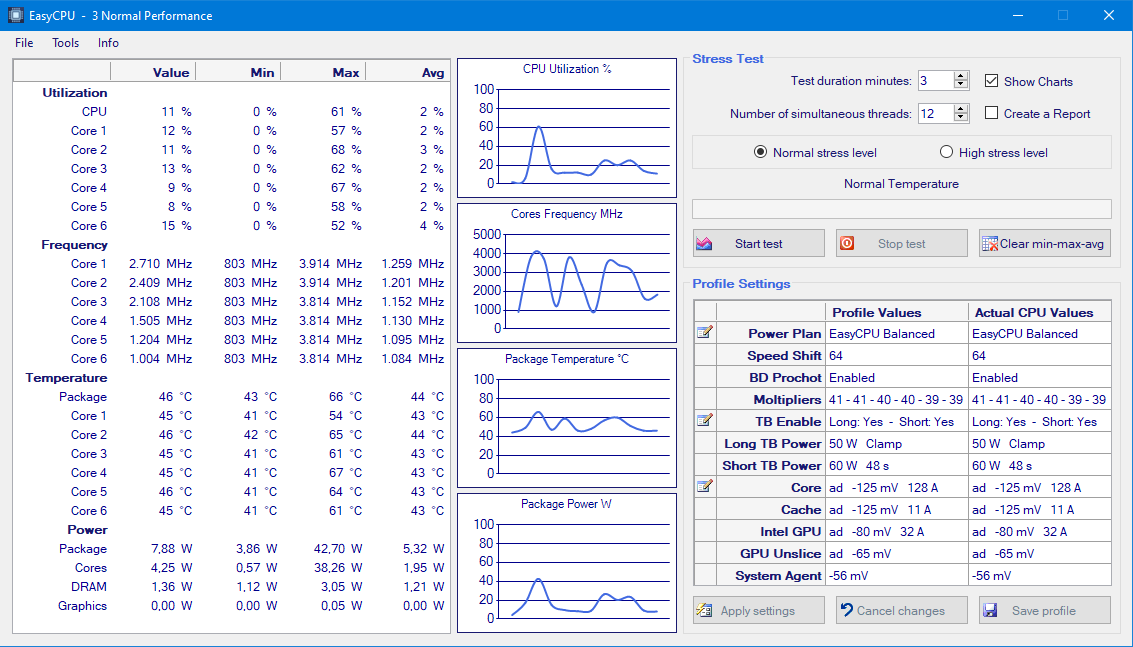
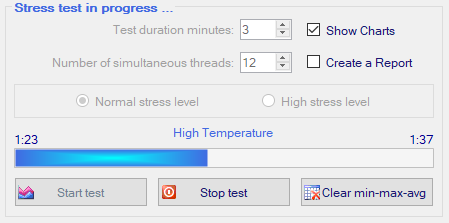
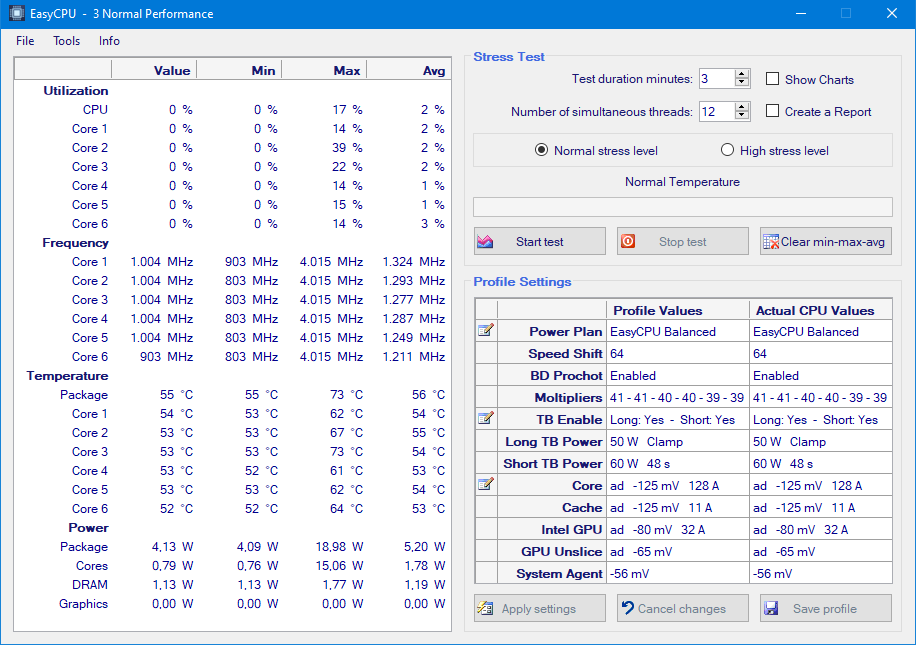
**Answer: -**

From this lab exercise, I have done some research on the online simulators that are mentioned above, “Little Man Computer” and “EasyCPU”, both seems to be very interesting as we will come to know how the central processing units are able to manage, send , retrieve data and so forth.

Below is the screenshot from LMC Simulator where a program is being load into the memory (either into system memory or RAM). Once the program is run in the central processing unit, It will tell us the number of program Counter, MAR, MDR, CIR, accumulator number with clock speed able to be defined. The result is the simulator will show the random(ly) accessed memory and current FDE cycle and Log File.



Similarly, for Easy CPU, below are the screenshots captured about starting a test, ending a test, testing results about frequency, utilization of the CPU, temperature of processor and so forth.



**References**

References: - [https://www.101computing.net/LMC/#](https://www.101computing.net/LMC/)

References: - <https://www.easy-cpu.com/>