First we had to familiar with Graphite multicore simulator tool, which is developed by Carbon group of MIT. We tried to install Graphite in many other operating systems. But it was compatible working only in Ubuntu 12.04 . Tool is generally used by people around the world for multicore simulation of exascale order computing using thousands of processors. But we had to simulate for making mobile clusters for computing. After getting familiar and exploring the tool, we made a tutorial on how to install, write and compile apps on graphite which could be used to analyse the results. In the initial stage, we used applications and benchmarks which are provided by Graphite for simulation. For setting up a simulation, there is a configuration file named “carbon_sim.cfg”, which contains what different configuration we need to set for the simulation. The different configurations are L1 and L2 Cache-size, RAM, Processor clock speed, Disk RPM, etc. After setting our required configuration, we have to compile and run the app on this environment. We could also define how many processor cores and process threads during the compile time of the application. Then these simulations would give the results in “result.out” file. Major parameters that matter for analysing the performance of the simulated environment are cache hit rate, cache miss rate, execution start time, execution end time, CPU utilization and throughput. In this way, we analysed the performance of currently trending mobile phones like Samsung S8 and iPhone 7 and compared them using graphical representation. Primarily we tested two types of applications on the simulated environments: high performance requiring apps like matrix multiplication, barnes benchmark test, etc. and moderate performance requiring apps like ping pong test, hello world test, etc. Then we made an automated script in python, which would take a whole set of already defined different configurations and test them on an app sequentially by itself. And later it would store the results in a folder.