

**CS 553 Cloud Computing
Programming Assignment 1
Design Document**

**Hariprasad Ravi Kumar
A20348609**

I have worked on the following three types of Benchmarks:

- CPU benchmarking
- Disk benchmarking
- Memory benchmarking

CPU Benchmarking:

I have written the code in C to measure the benchmarks.

The main program consists of 2 functions one which calculates the time for floating point operations and the other for integer operations. The loop runs till the maximum value of the integers. These values are then sent to threadFunctionFlop() and threadFunctionlop function to calculate the values in term of Giga. The values of other instructions that take place in the loop (the loop execution, loop increment, loop condition checking, the assignment operation, addition, loop end condition).

For multi-threading, have used the pthread library. Each of the threads would calculate the FLOPS and IOPS one after the other. I have automated the running of the program which will run 1,2,4, threads as per user choice and display the results using the flops() and iops() functions.

As a separate experiment, I have used the same benchmark on floating point and integer instructions and 4 threads for a 10-minute period for each one, and took samples every second on how many instructions per second were achieved during the experiment to files called 600samplesFLOPS.txt and 600samplesIOPS.txt.

Disk Benchmarking:

The design of program includes four different functions for each block size.

The functions for each block size (i.e. 1 Byte, 1 KB, 1 MB) are taken for Sequential and Random.

In both Sequential and Random i have read and write, so there are 4 functions Random read, Random write, Sequential read, Sequential write.

In random access the read and write operations are random whereas in sequential they are in an order.

Throughput and latency are calculated.

Memory Benchmarking:

I have written the memory benchmarking code in C language.

For varying size of blocks i.e. for 1 Byte, 1 KB and 1 MB i have written the sequential and random access.

I have used memcpy which executes 2 operation 1 read and 1 write operation.

For multithreading, have used the pthread library.

Throughput is calculated by finding the number of read/write operations performed per second by a particular block size.

To find the latency, i need to execute a single read/write for some amount of time and divide this time by number of iterations and the block size.

Improvements and Extensions to program:

The possible improvements and extensions to our program are as follows:

CPU benchmark:

I have used C for writing the benchmark. The gcc executes and results the appropriate output to file.out. The results can be improved by using other languages like JAVA and C++.

Memory Benchmark:

I have used memcpy() function to test this benchmark. I can also use alternative methods to test such as swap pointers only, not the data itself etc.

Isolate the effects of cache.

Disk Benchmark:

The level of concurrency may be increased.

These benchmarks are executed in Virtual Box, due to which there is some variation in the values. These programs can be executed in other environments also.

Since I'm running the virtual machine only one core, the rate and amount of data transfer is limited to this core. I can extend it to a multi core environment to get more accurate results.

I can extend our project to find the rate of data transfer in virtual disk through a virtualization.

As there are possibilities of bad sectors in any disk, due to which latency increases. I need to keep a check on these bad sectors.