Design Documentation

This programming assignment consists of 4 parts,

1. CPU Benchmark
2. Memory Benchmark
3. Disk Benchmark
4. Network Benchmark

The basic design of all the benchmarks are as follows:

(a) Functions are specified that does all the operations like IOPS and sequential read.
(b) Each benchmark runs in 4 levels of concurrency, 1, 2, 4, and 8 threads, respectively.
(c) Before starting pthread_create(), the time is reminded using gettimeofday() function, then the time will be reminded again after the execution. The difference of the time execution is used for further calculation.

The design description of all benchmarks:

1. **CPU Benchmark**: This has 3 separate Programs. The first program is to compute the GFLOPS and GIOPS which is implemented by doing multiple variables in float and integer type with elementary arithmetic operations. The second program is also to compute GFLOPS and GIOPS but using AVX instructions. The third program is to do integer and float operations for 10 minutes and get 600 samples for IOPS and FLOPS, which are generated every second.
2. **Memory Benchmark**: The program consists of 3 operation, sequential read+write, sequential write access and random write access using 8B, 8KB, 8MB and 80MB block sizes, using memcpy() and memset, respectively.
3. **Disk Benchmark:** The program consists of 3 operations, sequential read+write, sequential read access and random read access using 8B, 8KB, 8MB and 80MB block sizes.
(a) I allocated a 10 GB binary file named "new_file.bin" using sequential read before all the three operations. All the operations were executed after creating the file. The experiment time are only counted for three operations (not including 10GB binary file allocating time).
(b) To make sure every experiment runs at least 10 seconds. I adjusted the operation data size while testing. The data size details and run time are denoted in performance table.To make sure using strong scaling, I keep fixed operation data size for each block size in a specific operation. And then change the concurrency to test the difference.To make sure I am testing the disk benchmark instead of using RAM. I disabled buffer before each operation so that they are executed directly on disk.
(c) Fopen(), fclose(), fwrite(), fread(), etc are used respectively. These are faster than open(), write(), read().
4. **Network Benchmark:** The program consists of two operations, TCP protocol network transfer and UDP protocol network transfer. The two operations were tested on loopback

interface. The program can be also applied to two nodes test. Due to the limited resource on Chameleon we did not test two nodes.

(a) Both TCP and UDP operations transfers ping-pong messages, fixed packet size of 64KB for each transferring. The latency is measured by round trip time(RTT). Multiple threads are applied on both server and client side. The data size we transfer for all 4 types of threads are fixed to ensure the strong scaling.

(b) Firstly, 25.6MB fixed data size (i.e. 200 round trips) were tested for both TCP and UDP operations. After that, 128MB fixed data size (i.e. 1000 round trips) and 8GB fixed data size (i.e. 62500 round trips) were tested for UDP operation. Larger data size of transferring is not applied due to extremely long running time. Results of UDP show throughput and latency don't vary a lot if changing transferring data size, since throughput and latency are unit performance measurement.

Improvements and Extensions to program:

1. **CPU Benchmark**:
    i.   Much more complex operations in AVX instruction can be used
    ii. More threads can be used
    iii. The 600-samples program should draw real-time graph to show the trends of the FLOPS and   IOPS as the time progresses

2.   **Memory Benchmark :**
    i.   More functions except memcpy() and memset() could be tried out and check whether the performance has better or not.
    ii. More threads can be used

3. **Disk Benchmark:**
    i. Better APIs of read and write to disk might be used.
    ii. More threads might be applied.
    iii. Might create a 10GB text file and repeat the operations to compare the performance difference between binary file and text file.

4. **Network Benchmark:**
    i. Two nodes test can be applied.
    ii. Single trip message transfer might be applied to compare its result with RTT.
    iii. Number of successfully data transferring might be counted.