CS 553 Cloud Computing

Programming Assignment 1

Design Document

Disk Benchmarking:

- Disk Benchmarking is programmed in Java Language.
- The program includes read+write operations, sequential access, random access operations. Specifically, it includes 5 methods for read+write operations, sequential write, sequential read, random write, random read operations.
- User is asked for an input for number of threads {1, 2, 4, 8} for the program execution.
- Program loops through the block size list having the 4 block sizes {8B, 8KB, 8MB, 80MB}.
- Inside the block size loop there is the thread loop which creates the number of threads specified by the user one after another.
- As the thread is created, it calls the run () method which executes all the 5 methods for implementing the operations.
- The total time for each operation is stored in a HashMap created for each of the 5 operations performed per block size. We get the this total time by adding the time difference which we get using NanoTime() for start and end timings.
- These timings are used to calculate the Throughput and Latency for each operation performed and will be saved in the text file.

Memory Benchmarking:

- Memory Benchmarking is programmed in C Language.
- The program includes read+write operations, sequential write access, random write access operations. Specifically, it includes 3 methods for read+write operations, sequential write, random write operations.
- User is asked for an input for number of threads {1, 2, 4, 8} for the program execution.
- Program loops through the block size list having the 4 block sizes {8B, 8KB, 8MB, 80MB}.
- Inside the block size loop there is the thread loop which creates the number of threads specified by the user one after another.
- Per thread a method is called which has the 3 methods for the specified 3 operations to perform.
- The total time for each operation is calculated per block size using the difference of clock start and end timings.
- These timings are used to calculate the Throughput and Latency for each operation performed and will be saved in the text file.

CPU Benchmarking:

- CPU Benchmarking is programmed in C Language.
- The program includes measuring the processor speed at various level of concurrency (1 thread,2 thread,4 thread,8 thread)
- In the main function we call four function which return GFlops and Iops for the number of threads specified and 600 samples for Flops and Iops measured every second.
- There is the thread loop which creates the number of threads specified by the user one after another and per thread it calls a function which performs specific AVX instructions and calculate the time required to perform those floating and integer instructions.
- These timing are used to calculate processor speed and will be write in the file.

Network Benchmarking

- Network Benchmarking is programmed in Java.
- The program includes measuring the Throughput and latency for TCP/UDP for a fixed packet/buffer size (64 kb) at various level of concurrency (1 thread,2 thread,4 thread,8 thread).
- After running TCP_Server.java, It will take an available port and will wait for the client to connect.
- TCP_client.java ask user for no of threads input and port number displayed by the server. Client than establish a TCP connection and send the data in fixed data buffer (64 KB).
- After running UDP_Server.java, It will take an available port and will wait for the client to connect.
- UDP_client.java ask user for no of threads input and port number displayed by the server. Client than establish a UDP connection and sends the data.
- Throughput and latency computation are done for all the above operations and save in a file.