Assignment 8 Report
Shantanu Gupta
1217434439

1. **System Configuration:**
    a. Number of cores: 6
    b. Core type: Physical
    c. Core frequency: 2.6 GHz
    d. Size of memory: 16GB
    e. Turbo Speed: 4.5 GHz
2. **Measured average time for one matrix multiplication:**
    a. Task 1: 4.167 s
    b. Task 2: 0.018 s
    c. Task 3: 0.017 s

As seen above, the average execution time of the hand-coded algorithm is maximum as it does each computation inside multiple loops. The spark implementation using the inbuilt internally using breeze as its linear algebra library performs a lot better on the task. The breeze implementation using netlib-java to load native linear algebra library performs comparably to the spark implementation with very little difference between the two.

Spark uses breeze as its linear algebra library, while breeze uses netlib-java to load native linear algebra library to compute operations. The performance between JVM and native linear algebra library (like BLAS) is significant. Here we compare three conditions:

- Java locally multiplies the two matrices on a worker.
- Spark using breeze to multiply the two matrices.
- Breeze locally multiply the two matrices on a worker.

The latter two tasks successfully load native BLAS, while the task1 one just uses JVM.

3. **Observations and learnings:**

I learned scala and spark coding basics. Setting up a spark project and using spark framework and BLAS libraries for fast iterative computations.
The spark and breeze implementations perform a lot better than the hand-coded version. Both successfully load native BLAS, which dramatically speeds up the computation.