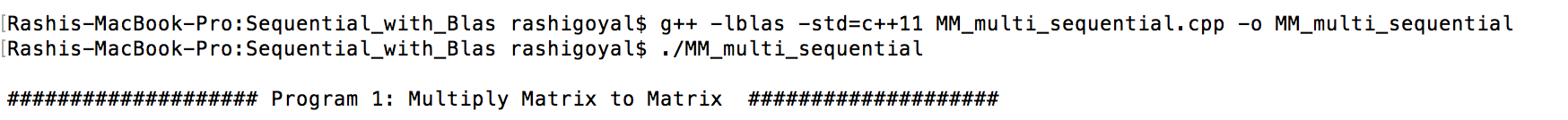
**Steps used to compile these programs:**

1. Compile the file using Terminal as shown in the following example

**g++ -lblas -std=c++11 MM\_multi\_threaded.cpp -o MM\_multi\_threaded**

1. Command mentioned in above step will create an output file
2. Run the output file using command ./ <FileName> for example

./

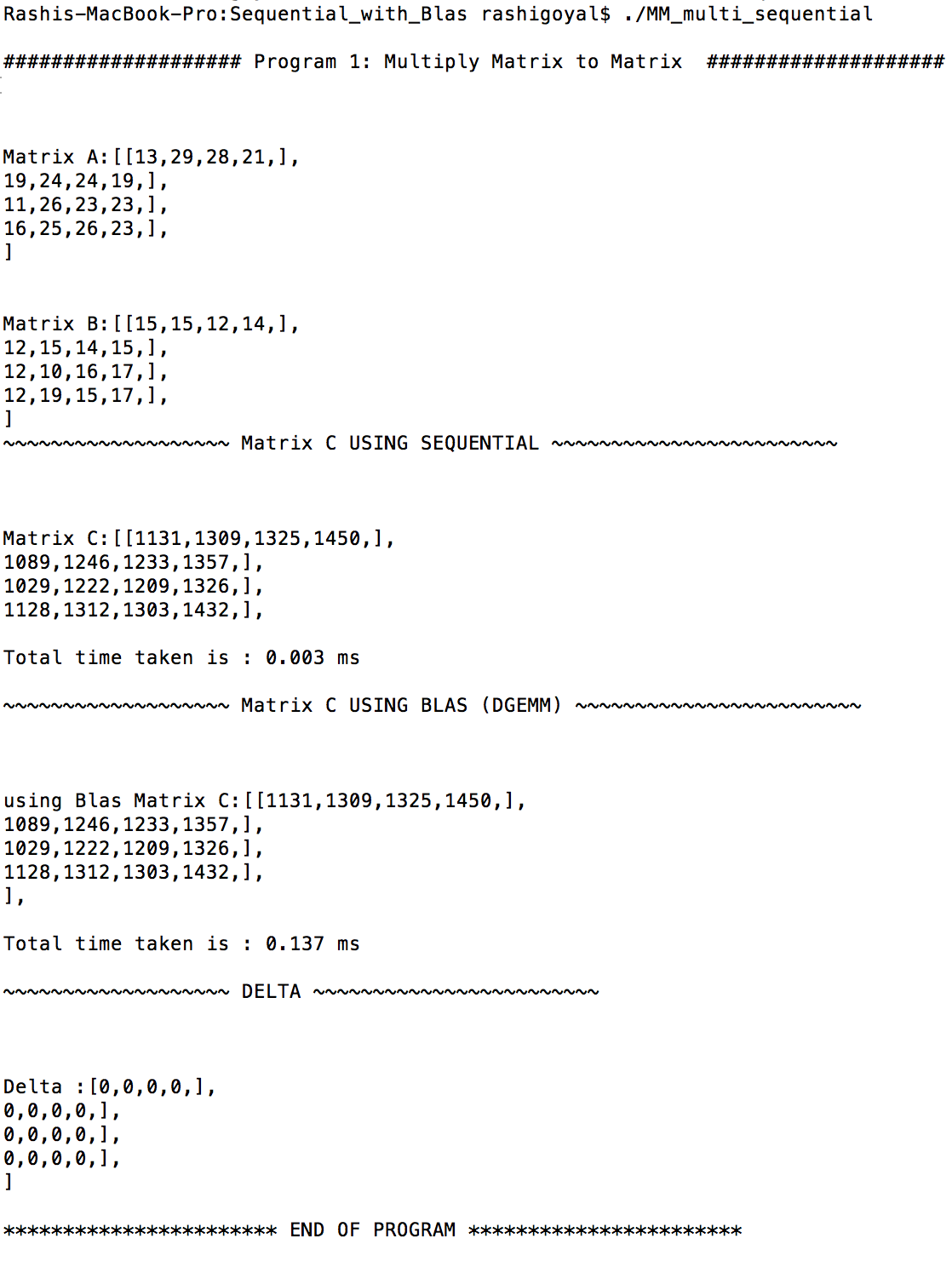
****

**Outputs:**

Please find screenshots for your reference. In these screenshots, First I have shown the values of the input Matrix and vectors. Second, I have shown the result matrix/vector using Sequential, BLAS and Threads respectively. Delta part of the output shows the difference between result calculated using BLAS and Sequential/Threaded solutions.

1. **Matrix-Matrix multiplication (Sequential and BLAS)**

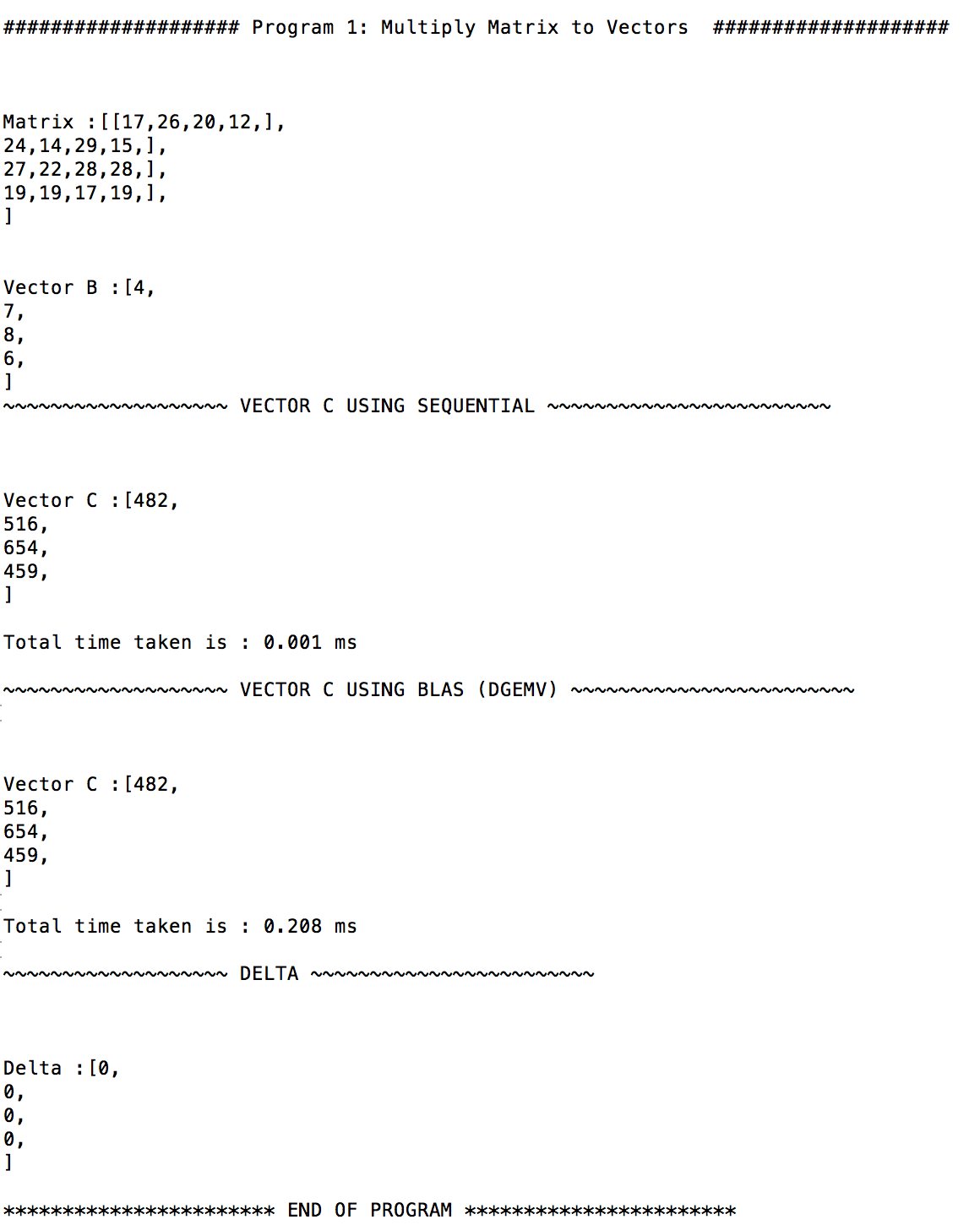
**Matrix Size: 4\*4**



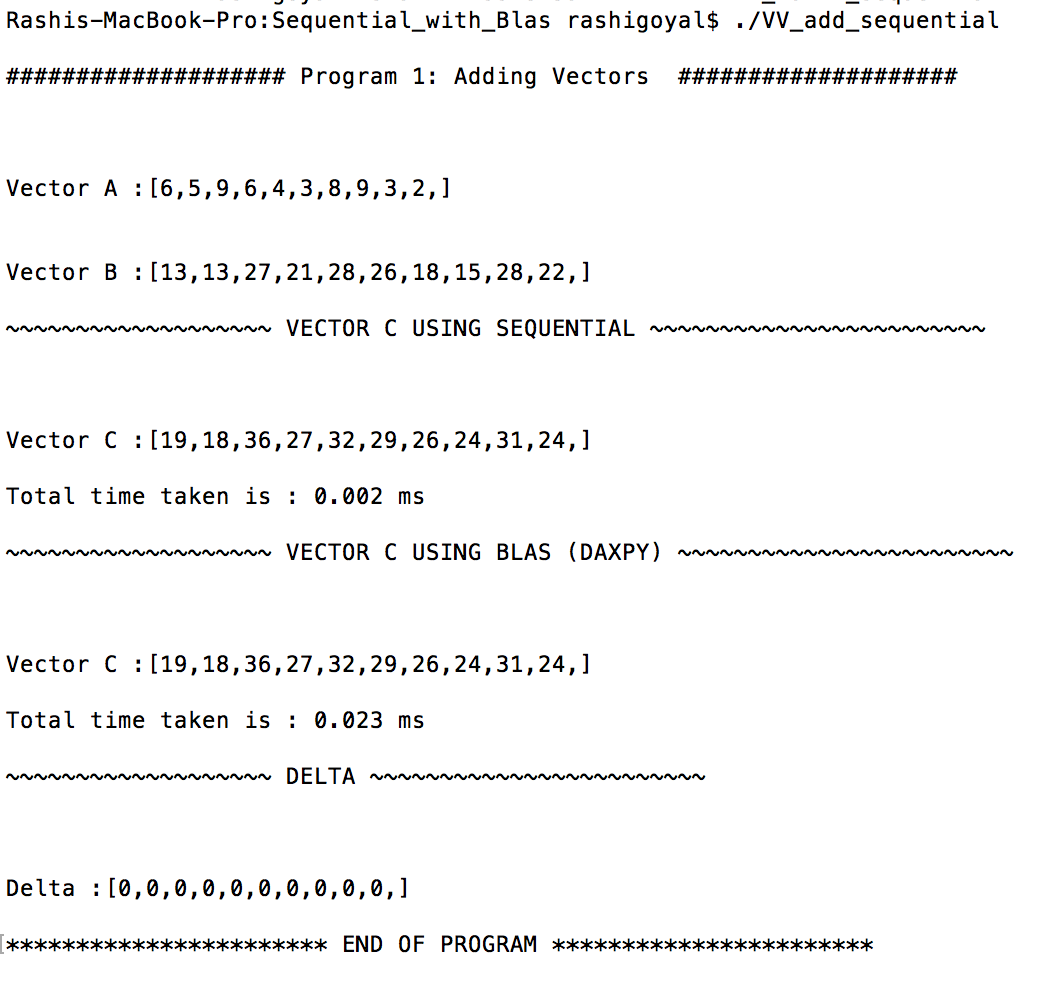
1. **Matrix vector Multiplication (Using Sequential and BLAS)**

**Matrix Size: 4\*4**

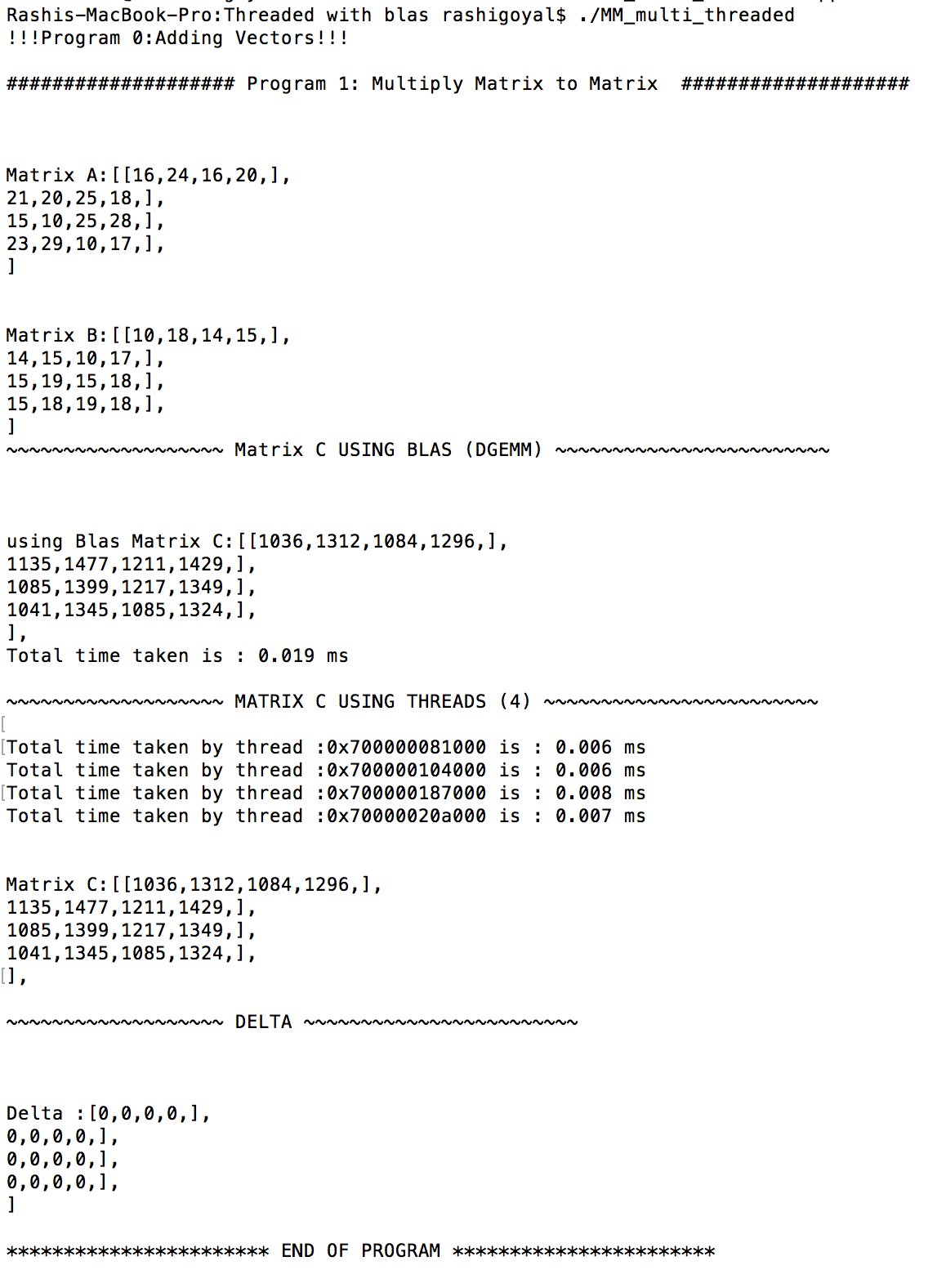
**Vector: 4\*1**



1. **Vector-Vector addition (Using Sequential and BLAS)**



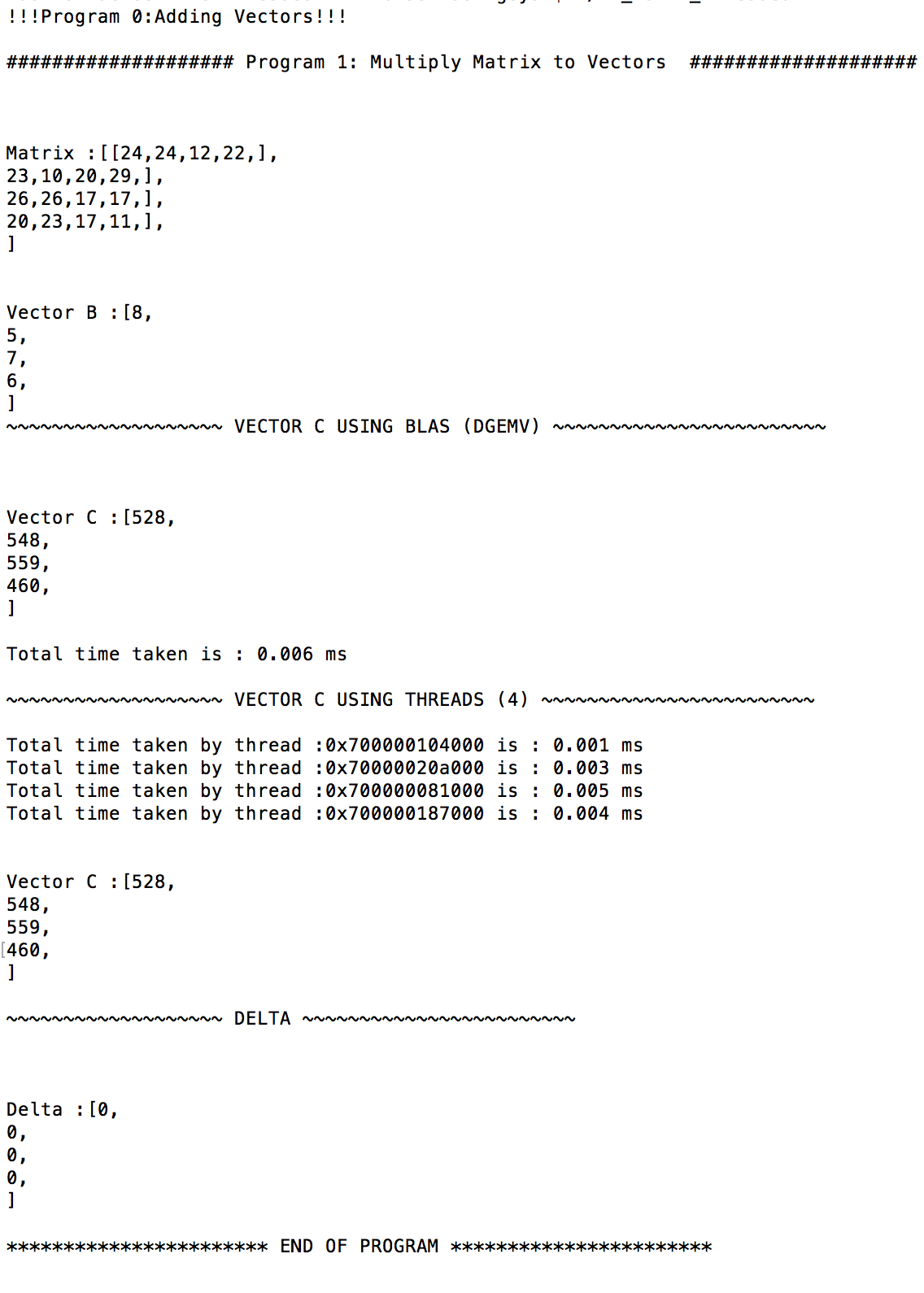
1. **Matrix-Matrix Multiplication (Using Threads and BLAS)**



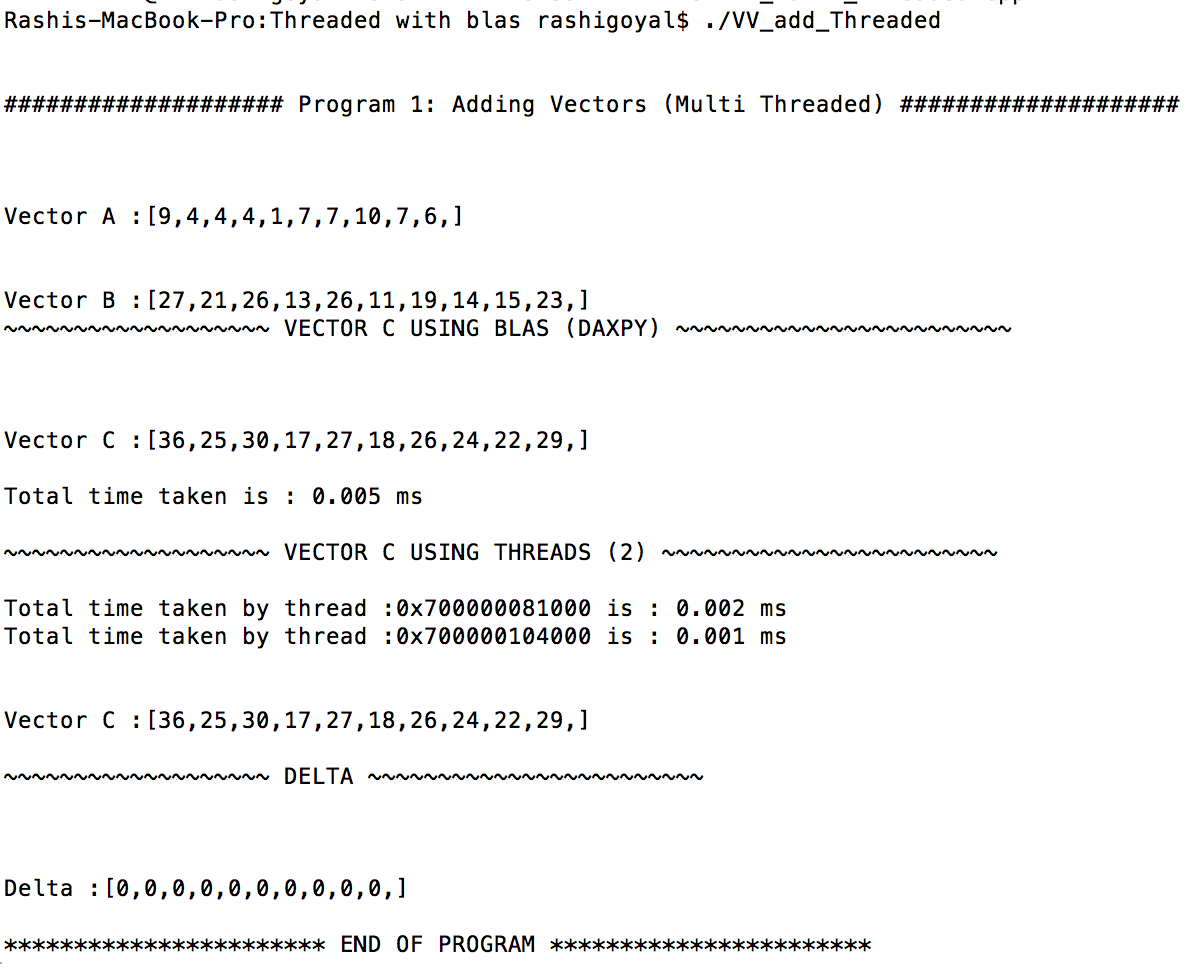
1. **Matrix vector Multiplication (Using Threads and BLAS)**

**Matrix Size: 4\*4**

**Vector: 4\*1**



1. **Vector-Vector addition (Using Threads and BLAS)**

****

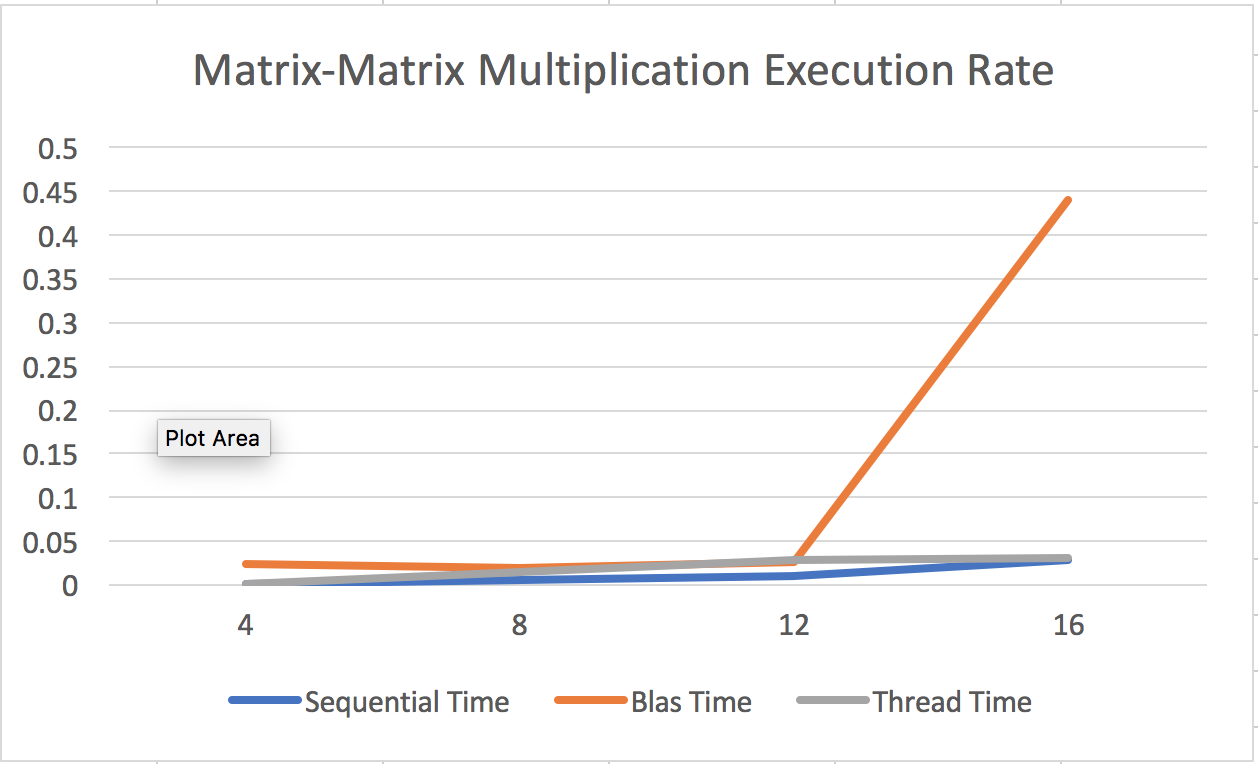
**Execution rate:**

There are three graphs displayed below. Each graph is a plot of time taken by the three processes for different size of matrix/vectors.

1. In below graph, x-axis represents the different matrix sizes for both input matrices where 4 means all matrix have 4\*4 elements and so on.

The y – axis represents time taken in ms.

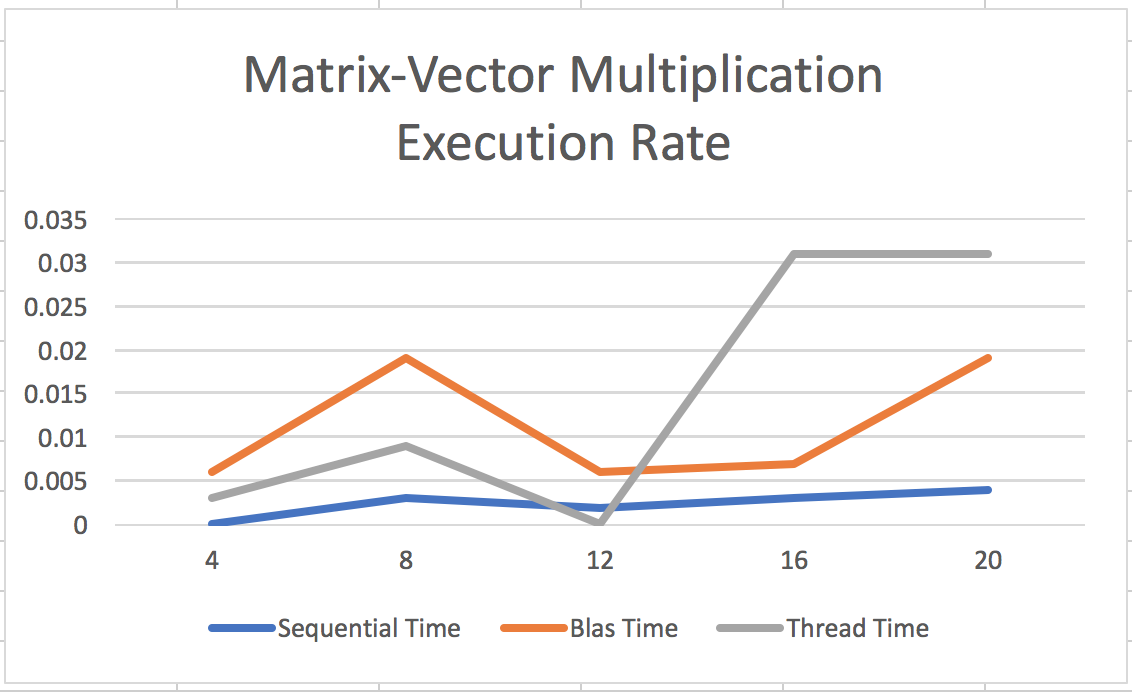
The key observation is BLAS time increased as the size of matrix increased whereas sequential and threaded approach did not show much variation.



1. In below graph, x-axis represents the different matrix and vector size for both input matrices where 4 means matrix has 4\*4 elements and vector has 4\*1 elements only and so on.

The y – axis represents time taken in ms.

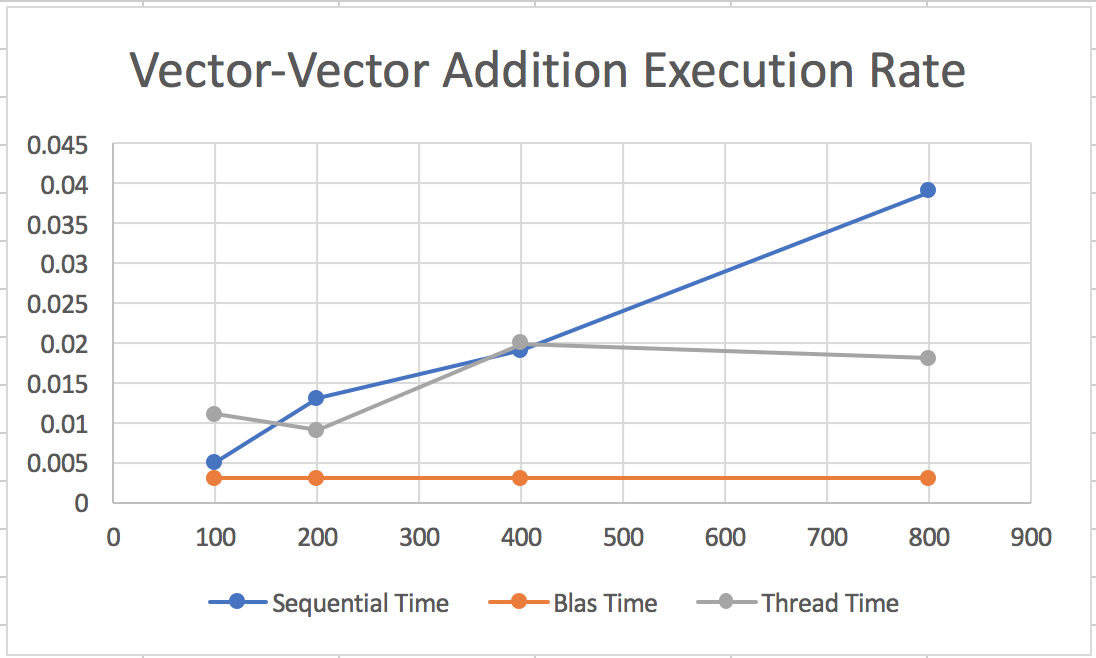
The key observation is both BLAS & thread implementation time varies as the number of input elements in matrix & vector increases whereas sequential approach did not show much variation.



1. In below graph, x-axis represents the different size for input vectors where 100 means vector has 100 elements and so on.

The y – axis represents time taken in ms.

The key observation is sequential implementation took most of the time where as BLAS was constant event if the number of elements raised up to 1000.



**System Specifications:**

