SYSC 4001 Assignment 1

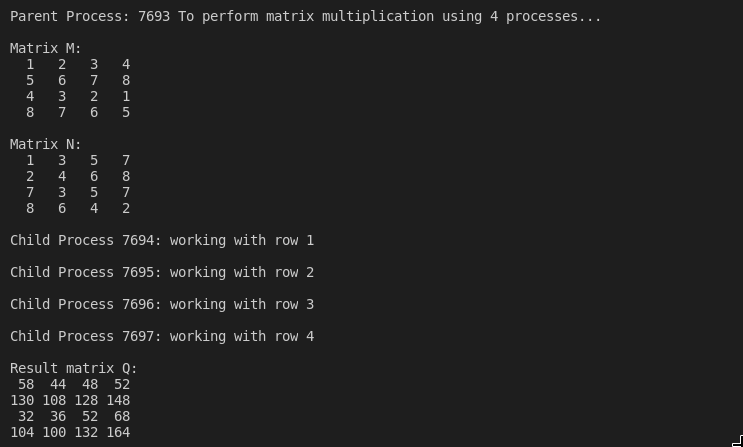
COncurrent Matrix Multiplication

Result Discussion

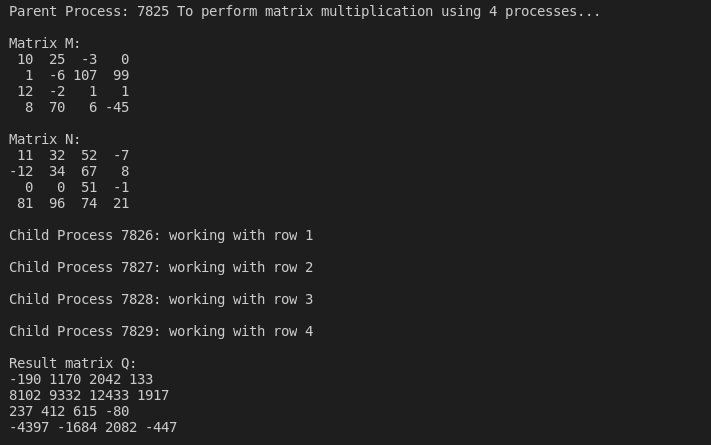
Part 1:

The program in part 1 MMULT1.c performs matrix multiplication using 4 processes, with each processes computing the result for one row in the result matrix.

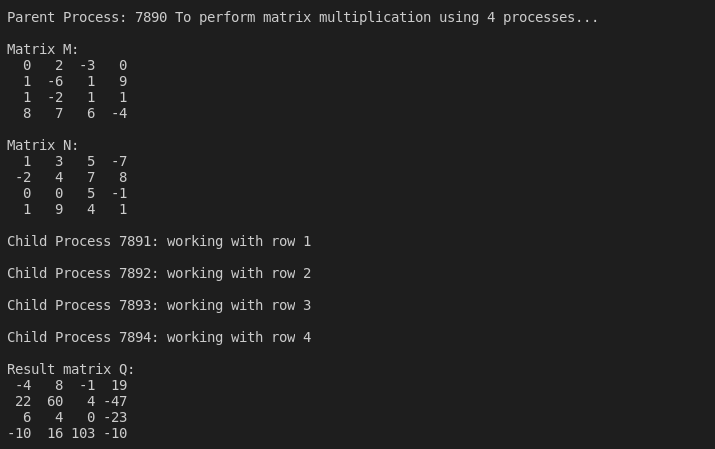
Test case 1 (mandatory test data): The first test case results are shown in the program output below



Test case 2: The second test case includes negative and zero and multi-digit integers



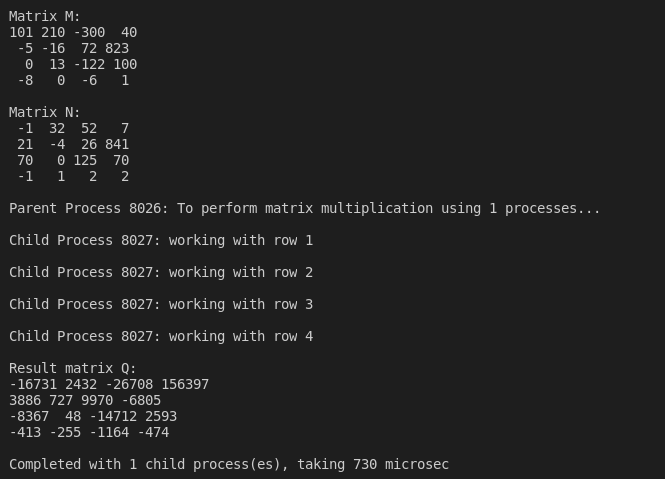
Test case 3:



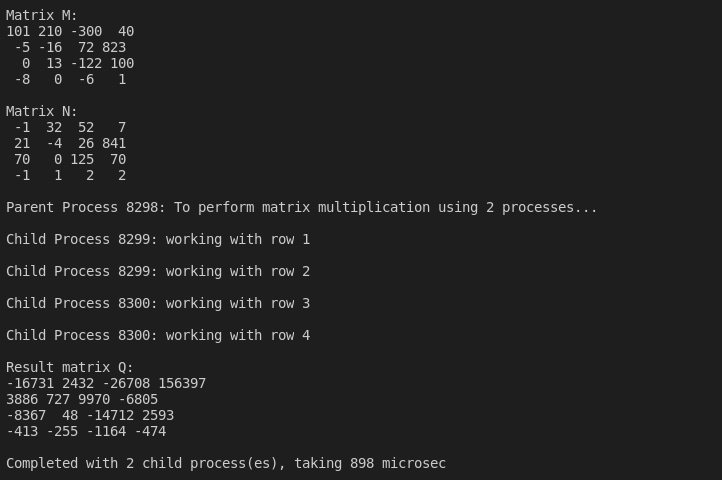
Part 2: The program in part 2 (MMULT2.c) modifies the program in part 1 to allow the user pass in a single argument (1, 2 or 4) which indicates the number of child processes to use to compute the resulting matrix. Each test case is run 3 times with 1, 2 and 4 as the arguments

Test case 1: Random numbers, including negative numbers and zero

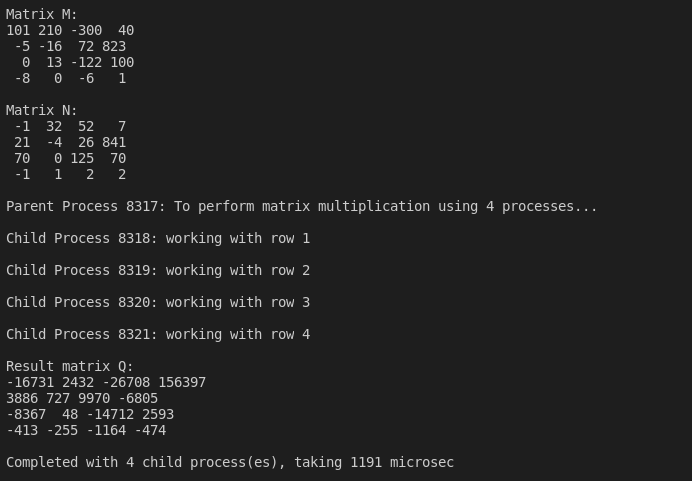
* 1 process specified



* 2 processes specified

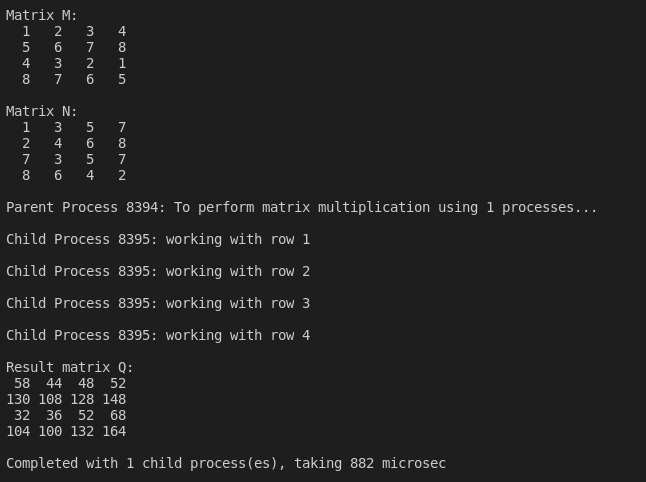


* 4 processes specified

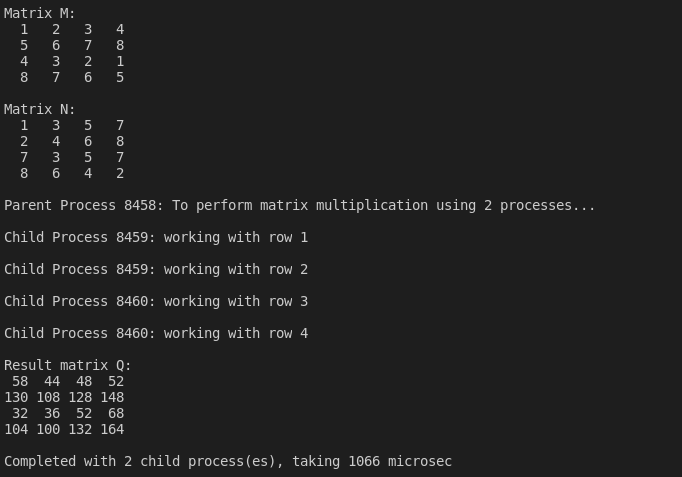


Test Case 2 (mandatory test data): Test data specified in the assignment specification, as before this case is run 3 times with 1, 2 and 4 as the argument

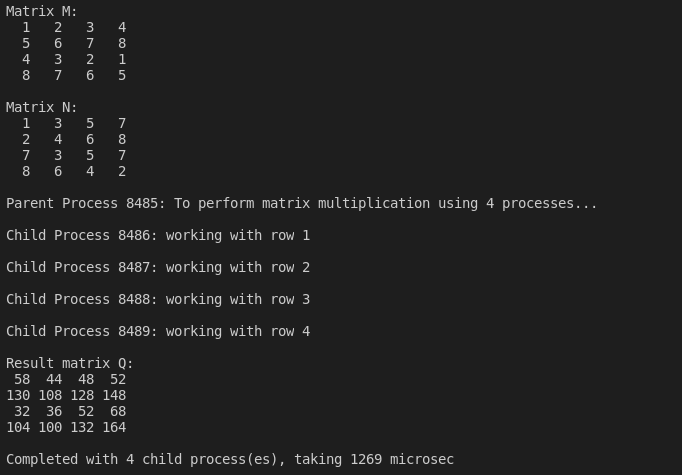
* 1 process specified



* 2 processes specified



* 4 processes specified



Computation time discussion:

For the second program, we observed that the more child processes we forked, the longer it took to perform the computation. This is likely as a result of the increased overhead of forking a child process being more than any productivity gains as a result of multiprocessing. This is as a result of the task being very computationally inexpensive.