Assignment 3 report

The number of concurrent processes is limited by multiple factors like how much memory is available to allocate to the new process, how many processes the cores can and are currently handling etc.

If the total available memory is say A bytes and each process (parent and child) is allocated say B bytes of memory for address space, then the maximum number of child processes possible will be at most A/B.

The total number of cores is 4 and each core can handle say C number of processes. Therefore, the total capacity to concurrently handle processes is 4C. Let's assume the total number of processes already being handled is D. Then the maximum number of child processes that can be run concurrently by the cores would be 4C-D. Therefore the maximum size of matrix that can be accommodated is r1*c2 = 4C-D. Having more processes may not be effective as no more cores are free.

We have tested the program for multiple values of r1*c2. The limit when new child process creation failed was approximately 7,000 children. So r1*c2 will be around 7000 for our case.