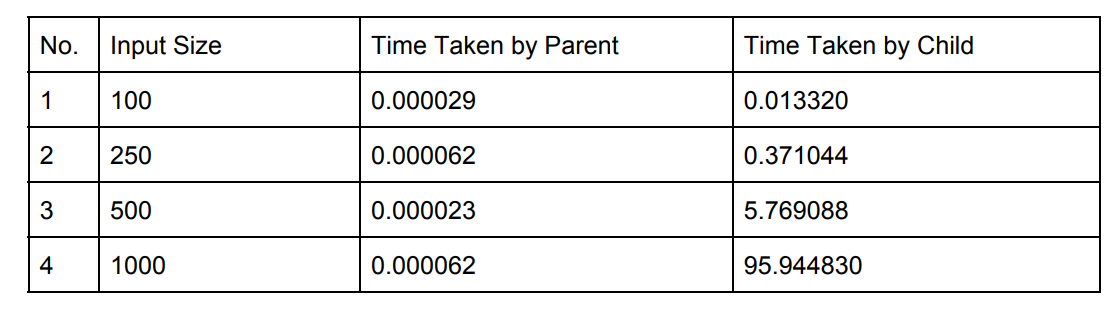
Addfour Program Description

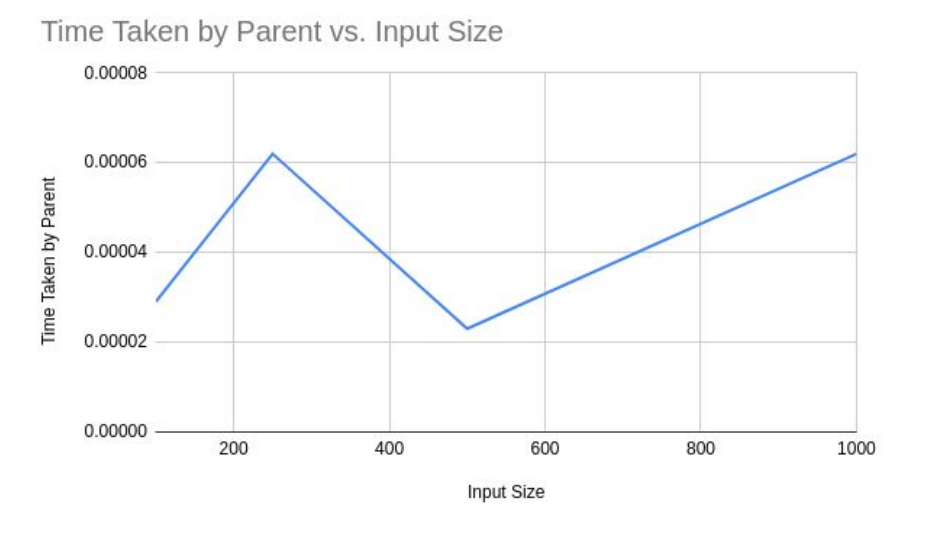
The main objective of the assignment is to illustrate that a child process needs its own address space using the four sum algorithm. The program takes a file through a command line argument, starts the child process through a fork system call, and starts the clock. The program then uses a while loop where it goes through the file of numbers and saves them to an array. Then, the program will go through four loops that will allow each number position to iterate through the file of numbers until all of the numbers have been added. Once this is finished, the clock and child process will be stopped and the time the child process took will be displayed. After this, the parent process will start, will use the wait function, and end. The time the parent process took to execute will be displayed after this, and then the program will exit.

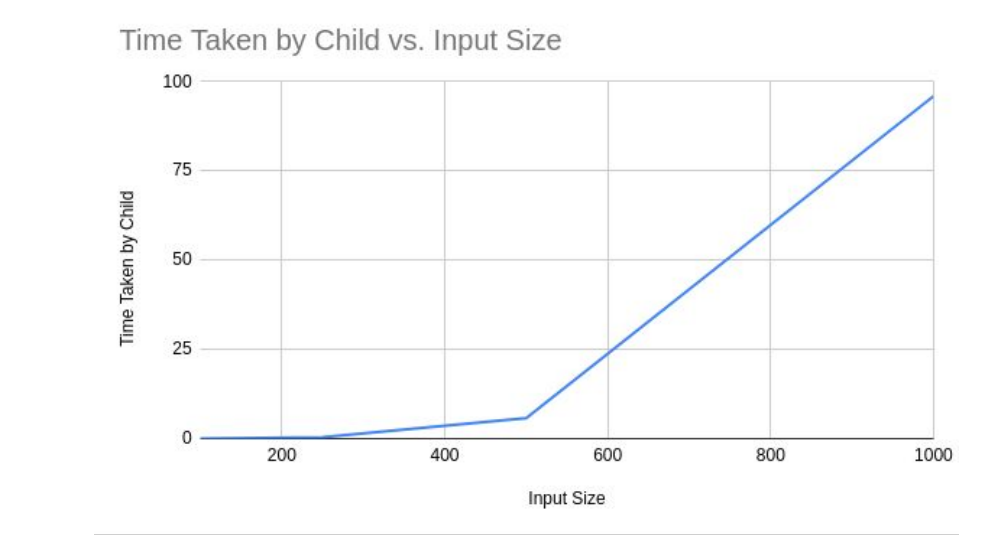
Results

The following table lists some of the results this program gave me with varying input sizes.



The following graphs detail the above results for both parent and child.



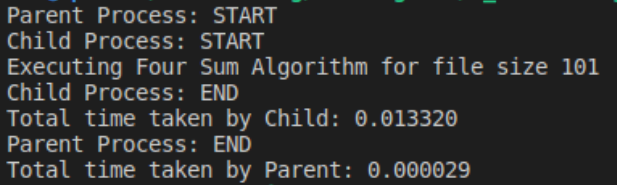


Explanation of Results

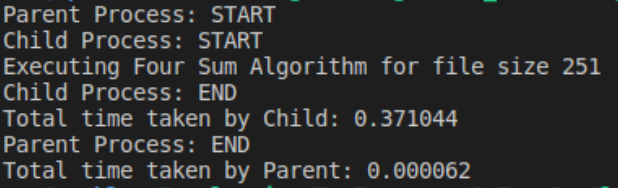
We can see through our results that the child process takes significantly more time than the parent process as the size of the input increases. This is because all of the computation is done by the child process, which occupies the CPU for most of the time, whereas the parent process executes almost close to zero time because it does not do any of the computations.

Terminal Screenshots

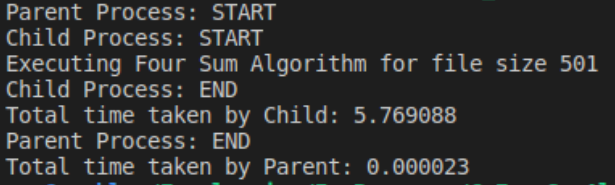
100 numbers:



250 numbers:



500 numbers:



1000 numbers:

