// Demetrios Doumas Task 2
// Process Management System call
// Interpretation of Part 2 in Task2

The code in part two has two children processes being created and two parent processes in total. The parent (a single main running program) creates the first child called q. If the parent successfully created the child q, then child q or process q creates another child called process R. Now process R has the Q process as a parent. If the q process failed to create the child process R, then it will execute its code.

There were several behaviors observed when running the code multiple times. There were no changes of the execution order. Every execution of the code gives the parent and children a new generated process ID. The fork() system call seems to work until the child is created and it terminates once its behavior has finished. After the fork() system call two process are running in parallel of the same code. The child has a copy of the parent's data in its own memory space. Any changes made to certain values in the parent process remains only in the parent process and same goes for the child process. The data manipulated in the child process stays in its own memory location that is completely separate from its parent. The value of b is originally 25. The parent process executes its behavior and b becomes 30. Once the first child q process is created, its parent is continuing working on executing the code at the same instance of its child. The child q has b equals to its original value of 25 and then changes to 45. Its parent process still has b equal to 30.

The child process q creates its own child called process R, so q becomes the parent of R. Now those two processes are running through the same code at an instance. The parent q runs its behavior and a has a value of 35 during its process. The parent of q has the value a still equal to 10. The new process R has its data copied from q, which results in b having its original value of 25 since it has not manipulated b in any way during its process. However, the value a is different and it is 905 for the R process. The q process has "a" equal to 35. The parent process still has the value of 10 for a.