The main difference between the Lottery and Round-Robin scheduling system is the priority that is used. In the round robin scheduler, all processes have the same priority, when the CPU is done with one process it will move to the next one and all will be given a certain amount of time to complete. While in lottery scheduling you can allocate more tickets to one process than another based on what is a higher priority. Round-robin scheduling provides fairness but may not be suitable for cases where some processes are more important than others. Lottery scheduling allows you to assign different priorities and, by manipulating tickets, control the relative amount of CPU time each process receives. It is good for prioritizing processes but is bad since it does require random implementation.

When the program runs it will first initialize a data structure of the processes that will be running in the CPU and will also reserve data for the tickets. Using p\_nice we will then assign lottery tickets according to the value of the p\_nice. Processes that have more tickets will have a higher priority than ones that will have less tickets. Using processSchedule() to randomly pick a ticket it will then choose what process to go and let that process run for however long it needs to. It will find the process by using the processWake() and processSuspend() to start other processes and stop others. When the process is done running the processSchedule() will then decrease the number of tickets the process and will go through a conditional if the process is lower a priority than other processes. Higher priorities will usually run longer than lower priorities. This will keep running until pcb is at zombie and will then exitSim()

When running the round robin DLXOS it was having the same time almost consistently, that is because since it goes through a round robin approach it will go through every single process and not use randomness to start another process and check priorities. The DLXOS that had a lottery scheduler would run longer than round robin usually. This is because since it must use randomness and check priorities this will take the system longer to go through compared to going through a round robin approach. But like previously stated, the lottery does have an advantage in certain circumstances. However when looking at the execution rate times for both these programs lottery is faster because when the higher priority programs get to the lower priority it will give less time to all processes and will end up like a round robin scheduler but will randomly choose which one will go next in the process run.