Ian Altoveros

CS433 Operating Systems

Professor Zhang

September 25th, 2020

This program is designed to create multiple objects of the PCB class. The PCB class has the known attributes of “ID, PRIORITY, STATE”. Each one with different given values, these values are then inserted into a table with a max size of 30 entries. The program is designed off the data structure of ReadyQueue, it will order the number of processes according to the priority given which is highest to lowest. To sort the table from highest to lowest I used a max heap function in the add function, the max heap or in my case heapify\_Up was made in a way where if the child node was bigger than the parent it would swap and recursively call the function once again to check if any other child nodes were bigger than the parent nodes. In the second test case the program will generate 15 random PCB objects with random ID’s and priorities, it will then sort the values depending on the priority.

I was unable to complete the second part of the 2nd test as I would get segmentation fault and I could not figure out why. I have made several iterations of the code trying to prevent the current size to fill up higher than the max size that was set but I was unable to figure out this portion of the homework assignment. I was only able to get the time clock running to remove a process by a random chance of 50%, it finished that in .80seconds.