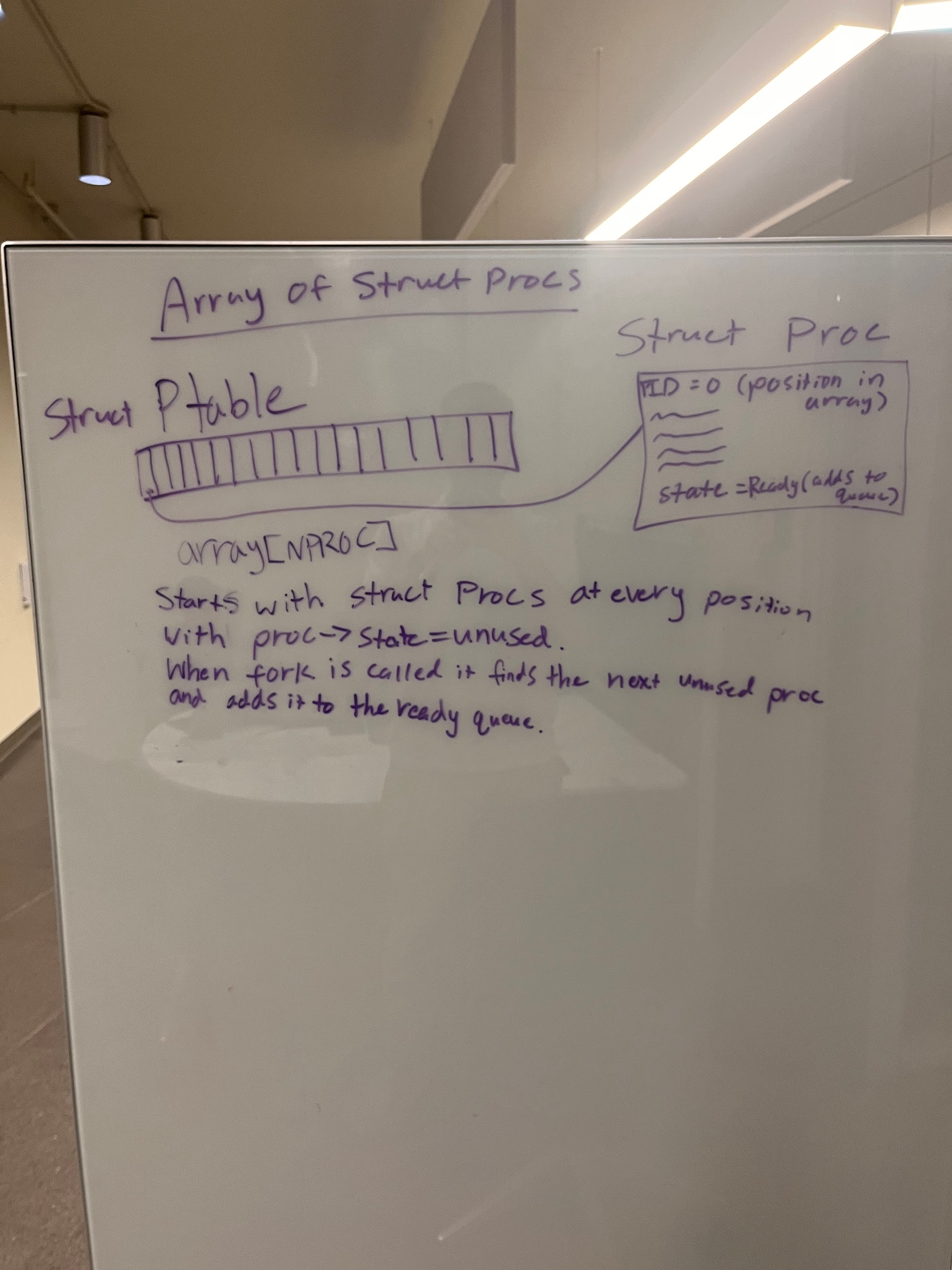
a.

b.

LCFS: This function starts by making sure the weights are correct and putting the current proc back in the ready queue. Then it searches through to find the proc with the smallest vruntime. This then makes the curr\_proc the smallest vruntime proc and changes it from ready to running. It then increments vruntime by the algorithm given. This algorithm is the 0 nice value weight divided by curr\_proc weight, times the timeslice. The timeslice is the curr\_proc weight divided by the total weight in the ptable times the scheduler latency. This ends up cycling through the procs evenly.

Lottery: This function uses tickets to determine priority of processes. A random number is generated based on the total tickets of all processes as the winning ticket number. The function searches through the array of all processes, adding up each of their tickets until the counter surpasses the winning ticket number. When the winner is found, it is changed to a running process. It is important to note that because a random number is being generated, the short term winners could all be the same process. Over the long run, the processes will run closer to their projected percentages.

c.

Files edited: proc.c, proc.h, defs.h, and main.c