OS Assignment

Student: Caio Marteli

Id: 19598552

Notes:

The first program Priority Scheduler uses a priority queue to serve as the ready queue.

The second program implements a similar type of queue but prioritises remaining burst time.

Important!

The thread scheduler is unfinished. I originally approached it by thinking I could run fully compiled programs A and B on threads however commands such as execl() replace the entire run time .  
By the time I realised the error in the approach I was already well out of time, hence my incomplete submission.

A separate docx named sourcecode is available, as requested.

Evidence of planning:

Notes used during conceptualisation of PP algorithm:

STEPS

SORT ARRAY BY A.T

ONCE LOADED CREATE A PRIORITY QUEUE (CALLED READY QUEUE) WITH ALL PROCESSES WITH A.T = 0

HAVE METHOD FOR CLOCK TICK THAT:

-CHECKS THE INPUT ARRAY FOR ANY ARRIVING PROCESSES -> ADDS THEM TO READY QUEUE

-ALSO subtracts BURST TIME FROM ACTIVE PROCESS

no need to sort list is using priority queue also arrival time is only used when first pushing to queue so it's in the heap it means it has already arrived

HAVE PROCESS METHOD THAT PRINTS THE LETTER OF RUNNING PROCESS AND SUBTRACTS FROM BURST TIME EVERY CLOCK TICK

IF A NEW PROCESS REACHES A.T CHECK IT PRIO IS HIGHER IF NOT LEAVE IT IN QUEUE

ONCE WE REACH END OF ARRAY STOP LOADING NEW PROCESSES INTO QUEUE

Testing evidence:

-------------------

Notes for SRJF algorithm:

use a priority but have the burst variable as act as priority



