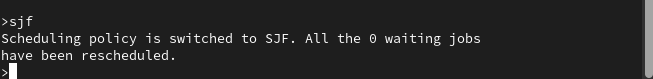
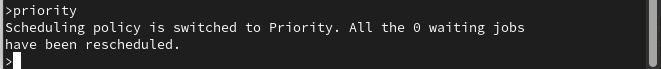
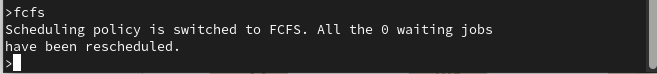
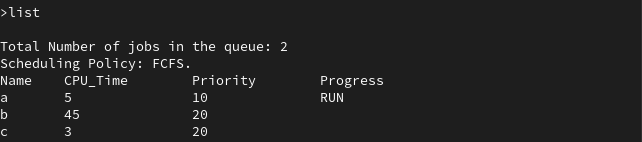
# Project 3 Report – A Batch Scheduling System by Sajith Muralidhar (szm0227)

Steps to run the program: -   
In your centos terminal, type the makefile command(The make fill create all executable files) run “make -f makerfile.txt”

Execute the program by./aubatch  
  
The output is as follows : -

1. Help: Text

   Description automatically generated
2. Run: - Text

   Description automatically generated with medium confidence
3. Sjf: - 
4. Priority: - 
5. Fcfs: -   
   Note: - if the scheduling policy is set and if we type the same command again, the prompt returns nothing. It only returns if the policy is changed.
6. List: -
7. Quit: -A screenshot of a computer

   Description automatically generated with medium confidence

System Design: -   
Diagram

Description automatically generated

Dataflow diagram: -

Diagram

Description automatically generated

Performance Metrics: -

Calculated turnaround time, CPU time, waiting time  
Passed 6 jobs in different scheduling algorithms.

Priority Scheduling: -  
Text

Description automatically generated

Shortest Job First: -   
Text

Description automatically generated

First Come First Serve: -   
Text

Description automatically generated with low confidence

Performance Evaluation: -

The waiting time of FCFS is higher than the other two, which makes the performance of FCFS the worst of the three. SJF had the quickest turnaround time and best performance followed by priority scheduling policy but priority being randomly assigned to each job, FCFS could have performed better. Arrival rate, number of jobs and processing time for each job are factors that determine the throughput(jobs/sec).

Lessons learnt: -

The project gives a clear picture on how scheduling algorithms function in coding or practically, where factors like PC performance effect in values from theoretical values. Also, this project taught me how to use pthread conditions to reduce system latency. Synchronization shows how it increases the performance of the system. Makefile makes it easier to execute bunch of files together and gives an order for execution of files.   
  
(Note: - we can remove all executable files after running the application by make -f makerfile.txt clean)