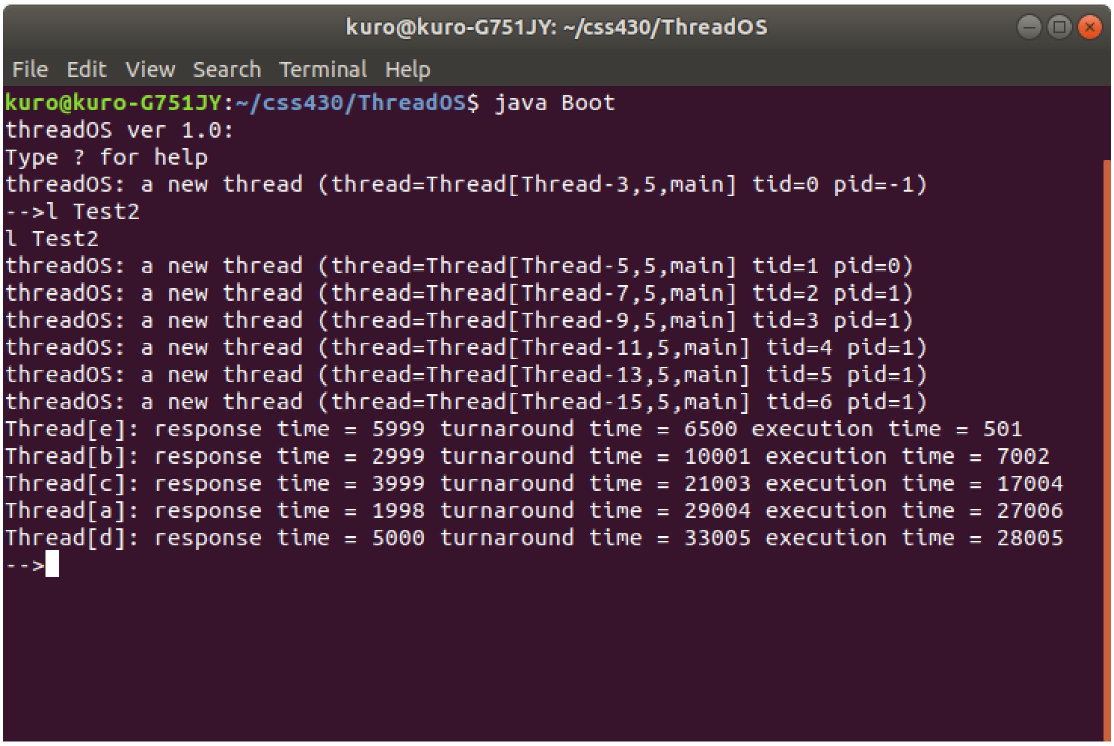
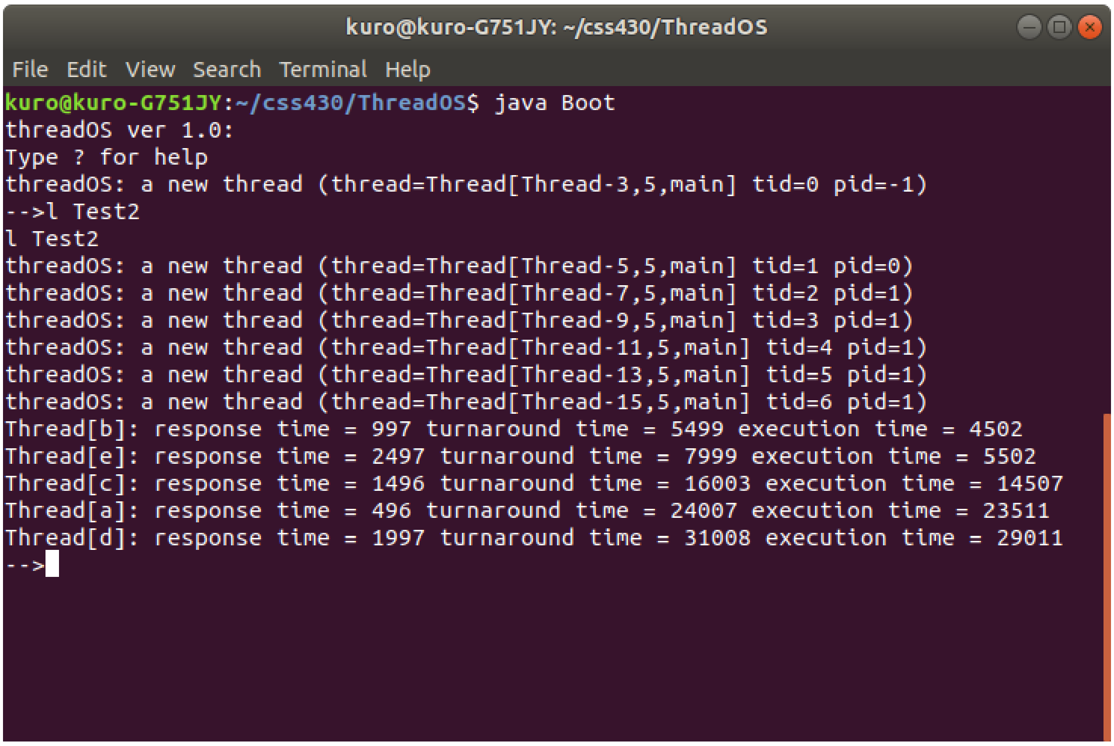
RR and MFQS comparison:

RR


kuro@kuro-G751JY: ~/css430/ThreadOS
File Edit View Search Terminal Help

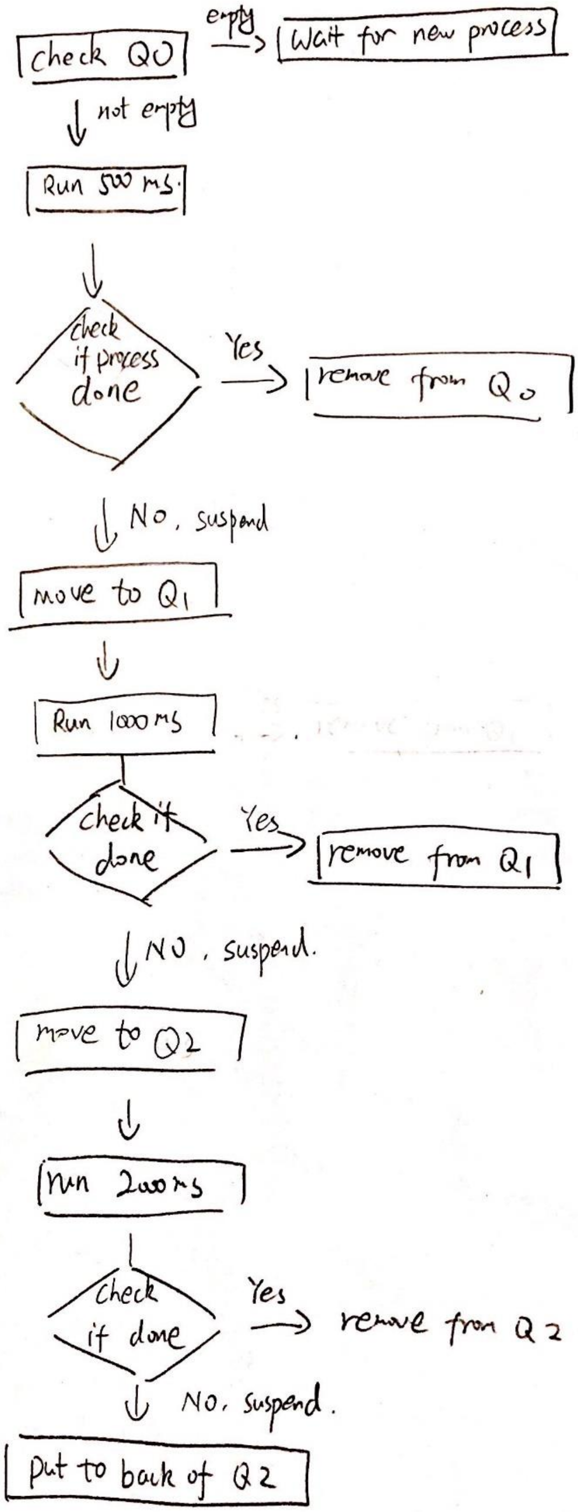
```
kuro@kuro-G751JY:~/css430/ThreadOS$ java Boot
threadOS ver 1.0:
Type ? for help
threadOS: a new thread (thread=Thread[Thread-3,5,main] tid=0 pid=-1)
-->l Test2
l Test2
threadOS: a new thread (thread=Thread[Thread-5,5,main] tid=1 pid=0)
threadOS: a new thread (thread=Thread[Thread-7,5,main] tid=2 pid=1)
threadOS: a new thread (thread=Thread[Thread-9,5,main] tid=3 pid=1)
threadOS: a new thread (thread=Thread[Thread-11,5,main] tid=4 pid=1)
threadOS: a new thread (thread=Thread[Thread-13,5,main] tid=5 pid=1)
threadOS: a new thread (thread=Thread[Thread-15,5,main] tid=6 pid=1)
Thread[e]: response time = 5999 turnaround time = 6500 execution time = 501
Thread[b]: response time = 2999 turnaround time = 10001 execution time = 7002
Thread[c]: response time = 3999 turnaround time = 21003 execution time = 17004
Thread[a]: response time = 1998 turnaround time = 29004 execution time = 27006
Thread[d]: response time = 5000 turnaround time = 33005 execution time = 28005
-->
```

MFQS


kuro@kuro-G751JY: ~/css430/ThreadOS
File Edit View Search Terminal Help

```
kuro@kuro-G751JY:~/css430/ThreadOS$ java Boot
threadOS ver 1.0:
Type ? for help
threadOS: a new thread (thread=Thread[Thread-3,5,main] tid=0 pid=-1)
-->l Test2
l Test2
threadOS: a new thread (thread=Thread[Thread-5,5,main] tid=1 pid=0)
threadOS: a new thread (thread=Thread[Thread-7,5,main] tid=2 pid=1)
threadOS: a new thread (thread=Thread[Thread-9,5,main] tid=3 pid=1)
threadOS: a new thread (thread=Thread[Thread-11,5,main] tid=4 pid=1)
threadOS: a new thread (thread=Thread[Thread-13,5,main] tid=5 pid=1)
threadOS: a new thread (thread=Thread[Thread-15,5,main] tid=6 pid=1)
Thread[b]: response time = 997 turnaround time = 5499 execution time = 4502
Thread[e]: response time = 2497 turnaround time = 7999 execution time = 5502
Thread[c]: response time = 1496 turnaround time = 16003 execution time = 14507
Thread[a]: response time = 496 turnaround time = 24007 execution time = 23511
Thread[d]: response time = 1997 turnaround time = 31008 execution time = 29011
-->
```

MFQS Algorithm:



Conclusion:

In general, MFQS scheduler preforms better than round robin scheduler. There are several points I want to mention.

1. MFQS has shorter response time since the first queue. Since its quantum is very short, it allows faster execution and accept next process.
2. MFQS has relative shorter Turnaround time. This shares the same reason above. First short queue allows process finish fast. So that process with small execution time can fast response and move out of the queue. In round robin scheduler, fast process need to wait in the queue for fix (1000ms) even it is already finished.
3. Notice some of execution time in MFQS is longer than RR. I think this because of the context switch.

FCFS:

If we use FCFS scheduler, it would cost much longer time. For example, thread a's CPU burst 5000ms. That means thread b need to wait 5000ms to execute. If we have some long CPU burst process comes first, the shorter process comes later will have a long response time.