Part 1

1. FIFO:  
   Job   0 -- Response: 0.00  Turnaround 1.00  Wait 0.00

  Job   1 -- Response: 1.00  Turnaround 4.00  Wait 1.00

  Job   2 -- Response: 4.00  Turnaround 12.00  Wait 4.00

SJF

Job   0 -- Response: 0.00  Turnaround 1.00  Wait 0.00

  Job   1 -- Response: 1.00  Turnaround 4.00  Wait 1.00

  Job   2 -- Response: 4.00  Turnaround 12.00  Wait 4.00

1. SJF(100)

Job   0 -- Response: 0.00  Turnaround 2.00  Wait 0.00

  Job   1 -- Response: 2.00  Turnaround 7.00  Wait 2.00

Job   2 -- Response: 7.00  Turnaround 15.00  Wait 7.00

SJF(200)

Job   0 -- Response: 0.00  Turnaround 1.00  Wait 0.00

  Job   1 -- Response: 1.00  Turnaround 4.00  Wait 1.00

  Job   2 -- Response: 4.00  Turnaround 12.00  Wait 4.00

SJF(300)

Job   0 -- Response: 0.00  Turnaround 1.00  Wait 0.00

  Job   1 -- Response: 1.00  Turnaround 4.00  Wait 1.00

  Job   2 -- Response: 4.00  Turnaround 12.00  Wait 4.00

FIFO(300)

Job   0 -- Response: 0.00  Turnaround 6.00  Wait 0.00

  Job   1 -- Response: 6.00  Turnaround 10.00  Wait 6.00

  Job   2 -- Response: 10.00  Turnaround 11.00  Wait 10.00

FIFO(200)

Job   0 -- Response: 0.00  Turnaround 1.00  Wait 0.00

  Job   1 -- Response: 1.00  Turnaround 4.00  Wait 1.00

  Job   2 -- Response: 4.00  Turnaround 12.00  Wait 4.00

FIFO(100)

Job   0 -- Response: 0.00  Turnaround 2.00  Wait 0.00

  Job   1 -- Response: 2.00  Turnaround 7.00  Wait 2.00

  Job   2 -- Response: 7.00  Turnaround 15.00  Wait 7.00

1. FIFO

Job   0 -- Response: 0.00  Turnaround 2.00  Wait 0.00

  Job   1 -- Response: 2.00  Turnaround 7.00  Wait 2.00

  Job   2 -- Response: 7.00  Turnaround 15.00  Wait 7.00

Part 2

1. Job  0: startTime   0 - runTime  84 - ioFreq   7

Job  1: startTime   0 - runTime  42 - ioFreq   3

1. I would prompt it to run as follows: ./mlfq.py --jlist 0,180,0:100,20,0 -Q 10,10,10
2. I would change how long the IO last using the –i flag. This would allow me to configure the scheduler parameters to behave just like a round robin scheduler.