Homework #3-4

1. Fork and Exec:

Run:

make args="arguments go here"

or

gcc -Wall -g -o hw3 hw3.c

./hw3.c arguments go here

2. Simple Scheduling:

a. FIFO Scheduling

	Start Time	End Time	Turnaround	Response
Job1	0	10	10	0
Job2	10	25	25	10
Job3	25	30	30	25
Job4	30	52	52	30

Avg. Turnaround = (10 + 25 + 30 + 52)/4 = 29.25 s

Avg. Response = (0 + 10 + 25 + 30)/4 = 16.25 s

b. SJF Scheduling

	Start Time	End Time	Turnaround	Response
Job1	30	55	55	30
Job2	15	30	30	15
Job3	0	5	5	0
Job4	5	15	15	5

Avg. Turnaround = (55 + 30 + 5 + 15)/4 = 26.25 s

Avg. Response = (30 + 15 + 0 + 5)/4 = 12.5 s

c. STCF Scheduling

	Start Time	End Time	Turnaround	Response
Job1	35	60	60	35
Job2	0	35	35	5
Job3	5	10	5	0
Job4	10	25	15	0

Avg. Turnaround = (60 + 35 + 5 + 15)/4 = 28.75 s

Avg. Response = (35 + 5 + 0 + 0)/4 = 10 s

d. RR Scheduling

	Start Time	End Time	Turnaround	Response
Job1	0	52	52	0
Job2	2	32	32	2
Job3	6	24	19	1
Job4	10	50	40	0

Avg. Turnaround = (52 + 32 + 19 + 40)/4 = 35.75 s

Avg. Response = (0 + 2 + 1 + 0)/4 = 0.75 s

3. More Complex Scheduling:

a. MLFQ Scheduling

	Turnaround	Response
Job 0	84	0
Job 1	153	7
Job 2	81	15
Job 3	70	18

Avg. Turnaround =
$$(84 + 153 + 81 + 70)/4 = 97$$
 s
Avg. Response = $(0 + 7 + 15 + 18)/4 = 10$ s

ii) In the range of 1 - 15, the average response time is shorter, but the average turnaround time is longer. After that point, the response time seems to stay right around 10 and the turnaround time goes over 100.

b. Lottery Scheduling

	Turnaround	Response
Job 0	13	2
Job 1	23	0
Job 2	18	1
Job 3	27	4

Avg. Turnaround =
$$(13 + 23 + 18 + 27)/4 = 20.25$$
 s
Avg. Response = $(2 + 0 + 1 + 4)/4 = 1.75$ s

ii) As the time slice time goes up, the turnaround and response times go up. If you put the slice time longer than the job's length, it still runs for the entire slice, so it can increase the times significantly if that is done.