

Operating Systems Project - Exercise 3

```
bence@BenceLaptop:~/OPsystems/Exercise 3$ gcc -o jobscheduling jobscheduling.c
bence@BenceLaptop:~/OPsystems/Exercise 3$ ./jobscheduling
please input job data file name: data.csv
output the origin job data
```

jobID	reachtime	needtime	privilege
1	800	50	0
2	815	30	1
3	830	25	2
4	835	20	2
5	845	15	2
6	700	10	1
7	820	5	0

FCFS job stream

jobID	reachtime	starttime	waittime	roundtime
6	700	700	0	10
1	800	800	0	50
2	815	850	35	65
7	820	880	60	65
3	830	885	55	80
4	835	910	75	95
5	845	930	85	100
total waiting time:310		total turnaround time:465		
average waiting time: 44.29		average turnaround time: 66.43		

SJF job stream

jobID	reachtime	needtime	starttime	waittime	roundtime
6	700	10	700	0	10
1	800	50	800	0	50
7	820	5	850	30	35
5	845	15	855	10	25
4	835	20	870	35	55
3	830	25	890	60	85
2	815	30	915	100	130
total waiting time:235		total turnaround time:390			
average waiting time: 33.57		average turnaround time: 55.71			

HRRF job stream

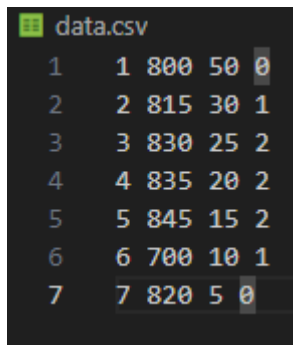
jobID	reachtime	responseratio	starttime	waittime	roundtime
6	700	0.00	700	0	10
1	800	0.00	800	0	50
7	820	6.00	850	30	35
2	815	1.33	855	40	70
5	845	2.67	885	40	55
4	835	3.25	900	65	85
3	830	3.60	920	90	115
total waiting time:265		total turnaround time:420			
average waiting time: 37.86		average turnaround time: 60.00			

HPF job stream

jobID	reachtime	privilege	starttime	waittime	roundtime
6	700	1	700	0	10
1	800	0	800	0	50
7	820	0	850	30	35
2	815	1	855	40	70
3	830	2	885	55	80
4	835	2	910	75	95
5	845	2	930	85	100
total waiting time:285		total turnaround time:440			
average waiting time: 40.71		average turnaround time: 62.86			

```
bence@BenceLaptop:~/OPsystems/Exercise 3$ |
```

The job information is printed out correctly as we can see from the output of this file:



1	1	800	50	0
2	2	815	30	1
3	3	830	25	2
4	4	835	20	2
5	5	845	15	2
6	6	700	10	1
7	7	820	5	0

FCFS runs correctly, and I have checked each calculation both in code and manually for its correctness.

For each of the next three algorithms, I used a modified version of the FCFS algorithm provided, a common function to modify the reached status after each iteration, and one supportive function based on the provided findminjob. After each algorithm I used the provided reset function with a bit of modification to make sure all data were back to original. For the printing in the algorithms, I provided an extra column (column 3) for the most important data for their calculations.

In the HRRF, it might look like not the highest response ratio was considered every time, but that is just because as the iterations progressed it grew, I manually checked to make sure it ran correctly.

I have included the data.csv file as well in the folder for testing purposes.