

# ICS Homework 14

June 14, 2020

## 1 Process Scheduling

We have following jobs in the workload. No I/O issues are involved.

Job	Arrival Time	Run Time
A	0ms	4ms
B	1ms	1ms
C	4ms	5ms
D	6ms	2ms

1. When a job arrives, it is added to the tail of the work queue.
2. CPU picks job to run after all queue operations.
3. The **MLFQ** policy has 2 priority queues, higher one with time-slice of 1ms and lower one with time-slice of 2ms. We use **RR** in each queue. Priority boost isn't supported.
4. No preemption in **MLFQ**.
5. We do RR by moving the **recently executed task** to the end of the queue.
6. The priority of operations is RR movement > acception new job.

Please calculate the **average** respond time and **average** turn-around time for different scheduling policies.

Scheduling Policy	Turnaround Time	Response Time
FIFO	[1]	[2]
STCF	[3]	[4]
MLFQ	[5]	[6]

## 2 Locking

What are problems in the following simple implementation of lock?

```
1 typedef struct __lock_t {  
2     int flag;  
3 } lock_t;  
4
```

```
5 void init (lock_t *mutex) {
6     mutex->flag = 0;
7     // 0 -> lock is available 1 -> held
8 }
9
10 void lock (lock_t *mutex) {
11     while (mutex->flag == 1) ;
12     // spin-wait (do nothing)
13     mutex->flag = 1;
14     // now SET it!
15 }
16
17 void unlock(lock_t *mutex) {
18     mutex->flag = 0;
19 }
```