



MIDTERM REVIEW

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.1

MIDTERM

- Thursday February 12th
- Meet in JOY 215 (2.0 hrs 3:40 - 5:40p)
- Test designed to take less than 2 hours
- Three pages of notes, double-sided, any-size paper permitted
- No book, other notes, cell phones, or internet
- Basic calculators OK
- Individual work
- Coverage: up through Chapter 27

- ***Preparation:***
- Practice quiz: Quiz 2: CPU scheduling (***posted***)
 - Auto grading w/ multiple attempts allowed as study aid
- Practice Midterm Questions-
 - Series of problems presented Tuesday
 - Solutions provided on Wednesday

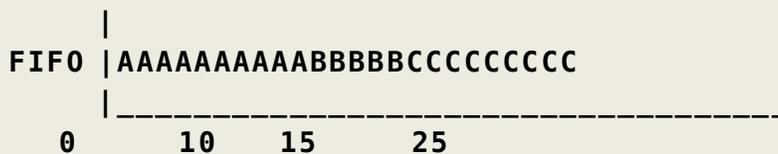
February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.2

FIFO EXAMPLE

- Operation of CPU schedulers can be visualized with timing graphs.
- The graph below depicts a FIFO scheduler where three jobs arrive in the sequence A, B, C, where job A runs for 10 time slices, job B for 5 time slices, and job C for 10 time slices.



February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.3

Q1- SHORTEST JOB FIRST (SJF) SCHEDULER

- Draw a scheduling graph for the SJF scheduler without preemption for the following jobs. Draw vertical lines for key events and be sure to label the X-axis times as in the example.

<u>Job</u>	<u>Arrival Time</u>	<u>Job Length</u>
A	T=0	25
B	T=5	10
C	T=10	15



February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.4

Q2 - STCF - 2

■What is the response time (RT) and turnaround time (TT) for jobs A, B, and C?

RT Job A: _____ TT Job A: _____

RT Job B: _____ TT Job B: _____

RT Job C: _____ TT Job C: _____

■What is the average response time for all jobs? _____

■What is the average turnaround time for all jobs? _____

February 10, 2026

TCS5422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.7

Q3 - OPERATING SYSTEM APIs

1. Provide a definition for what is a blocking API call
2. Provide a definition for a non-blocking API call
3. Provide an example of a blocking API call.
Consider APIs used to manage processes and/or threads.
4. Provide an example of a non-blocking API call.
Consider APIs used to manage processes and/or threads.

February 10, 2026

TCS5422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.8

Q4 – OPERATING SYSTEM APIs - II

1. When implementing memory synchronization for a multi-threaded program list one **advantage** of combining the use of a condition variable with a lock variable via the Linux C thread API calls: `pthread_mutex_lock()` and `pthread_cond_wait()`

2. When implementing memory synchronization for a multi-threaded program using locks, list one **disadvantage** of using blocking thread API calls such as the Linux C thread API calls for: `pthread_mutex_lock()` and `pthread_cond_wait()`

3. List (2) factors that cause Linux blocking API calls to introduce **overhead** into programs:

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.9

Q5 – PERFECT MULTITASKING OPERATING SYSTEM

In a perfect-multi-tasking operating system, every process of the same priority will always receive exactly $1/n^{\text{th}}$ of the available CPU time. Important CPU improvements for multi-tasking include: (1) fast context switching to enable jobs to be swapped in-and-out of the CPU very quickly, and (2) the use of a timer interrupt to preempt running jobs without the user voluntarily yielding the CPU. These innovations have enabled major improvements towards achieving a coveted “Perfect Multi-Tasking System”.

List and describe two challenges that remain complicating the full realization of a Perfect Multi-Tasking Operating System. In other words, what makes it very difficult for all jobs (for example, 10 jobs) of the same priority to receive **EXACTLY** the same runtime on the CPU? Your description must explain why the challenge is a problem for achieving perfect multi-tasking.

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.10

Q5 - PERFECT MULTITASKING OPERATING SYSTEM

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.11

Q6 - ROUND-ROBIN SCHEDULER

Show a scheduling graph for a Round-Robin (RR) scheduler with job preemption where newly arriving jobs will immediately run. Assume a time slice of 3 timer units. Draw vertical lines for key events and be sure to label the X-axis times as in the example.

Job	Arrival Time	Job Length
A	T=0	25
B	T=5	10
C	T=10	15



February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.12

Q6 - RR SCHEDULER - 2

Using the graph, from time $t=10$ until all jobs complete at $t=50$, evaluate Jain's Fairness Index:

Jain's fairness index is expressed as:

$$\mathcal{J}(x_1, x_2, \dots, x_n) = \frac{(\sum_{i=1}^n x_i)^2}{n \cdot \sum_{i=1}^n x_i^2}$$

Where n is the number of jobs, and x_i is the time share of each process Jain's fairness index=1 for best case fairness, and $1/n$ for worst case fairness.

For the time window from $t=10$ to $t=50$, what percentage of the CPU time is allocated to each of the jobs A, B, and C?

Job A: _____ Job B: _____ Job C: _____

With these values, calculate Jain's fairness index from $t=10$ to $t=50$.

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.13

Q6 - II

$$\mathcal{J}(x_1, x_2, \dots, x_n) = \frac{(\sum_{i=1}^n x_i)^2}{n \cdot \sum_{i=1}^n x_i^2}$$

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.14

Q7A – STRIDE SCHEDULER

Complete the timing table for the stride scheduler →

The first 6 time slices are already completed.

Answer the questions below the timing table.

Event	Pass (A) (stride=5)	Pass(B) (stride=20)	Who Runs (circle answer)?
1.	0	0	A (random selection)
2.	5	0	B
3.	5	20	A
4.	10	20	A
5.	15	20	A
6.	20	20	A (random selection)
7.	_____	_____	A B random
8.	_____	_____	A B random
9.	_____	_____	A B random
10.	_____	_____	A B random
11.	_____	_____	A B random

After the first 6 events, **how many scheduling actions** select job A to run? _____

select job B to run? _____

After 10 events, **what percentage of scheduling actions** does job A receive? _____

does job B receive? _____

Q7B – LOTTERY SCHEDULER

Job A holds tickets 0-49, Job B holds tickets 50-99. The lottery scheduler picks a random number (0-99) to select a job to run for each time slice.

Here are the first 20 winning tickets. Time slices numbers are shown below the ticket number.

44 81 11 16 8 77 97 17 47 14 99 61 87 49 68 2 24 75 53 48
 t_0 t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14} t_{15} t_{16} t_{17} t_{18} t_{19}

After 5 time slices (t_0 to t_4), **how many time slices** does job A receive? _____

does job B receive? _____

After 20 time slices (t_0 to t_{19}), **what percentage of time** does job A receive? _____

does job B receive? _____

Jackson deploys a 3-level MLFQ scheduler. The time slice is 1 for high priority jobs, 2 for medium priority, and 4 for low priority. This MLFQ scheduler performs a Priority Boost every 6 timer units. When the priority boost fires, the current job is preempted, and the next scheduled job is run in round-robin order.

Job	Arrival Time	Job Length
A	T=0	5
B	T=0	10
C	T=0	15

(11 points) Show a scheduling graph for the MLFQ scheduler for the jobs above. Draw vertical lines for key events and be sure to label the X-axis times as in the example. Please draw clearly. An unreadable graph will lose points.



Q8 PART B – MLFQ SCHEDULER

- Using the graph, calculate the turnaround time for each job, and the average turnaround time:
 - TT Job A: _____ AVG TT: _____
 - TT Job B: _____
 - TT Job C: _____

- Calculate the response time for each job, and the average response time:
 - RT Job A: _____ AVG RT: _____
 - RT Job B: _____
 - RT Job C: _____

Q9 - ROUND ROBIN SCHEDULER

- Consider a round-robin (RR) scheduler where upon new job arrival, the scheduler does not perform a context switch, but places the new job at the back of the RR queue. When the current job finishes its time quantum, a context switch is performed to execute the next job in the RR queue.
- The RR-scheduler has a 3-sec time slice.
- Draw a scheduling graph for the following jobs.

Job	Arrival Time	Job Length
A	T=0	10 seconds
B	T=2	12 seconds
C	T=10	8 seconds
D	T=16	6 seconds

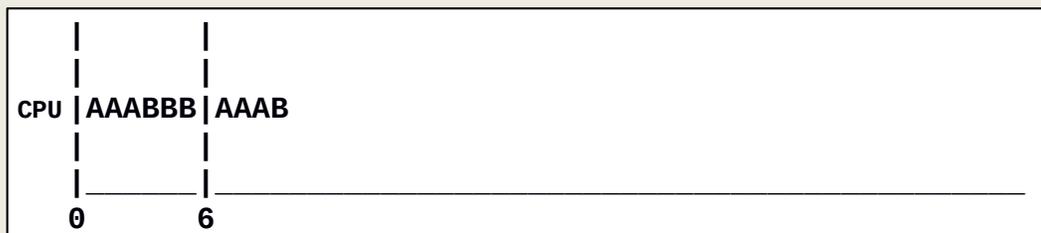
February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.19

Job	Arrival Time	Job Length
A	T=0	10 seconds
B	T=2	12 seconds
C	T=10	8 seconds
D	T=16	6 seconds

Starting Graph:



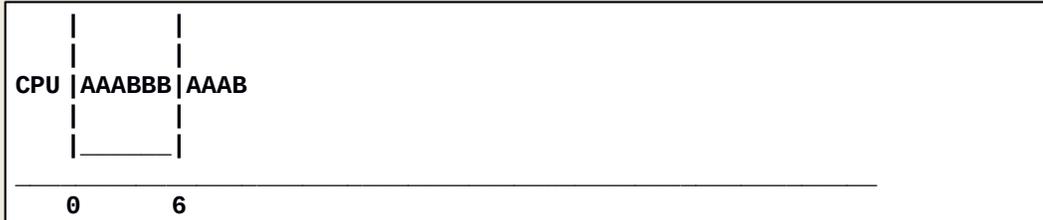
February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

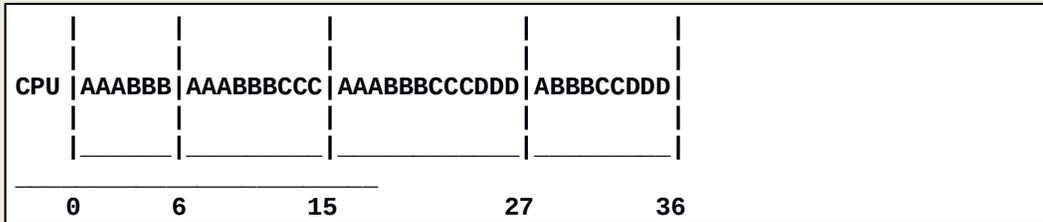
L10.20

Job	Arrival Time	Job Length
A	T=0	10 seconds
B	T=2	12 seconds
C	T=10	8 seconds
D	T=16	6 seconds

Starting Graph:



Solution:



Q9 PART B – ROUND ROBIN SCHEDULER

- Using the graph, calculate the turnaround time for each job, and the average turnaround time
- TT Job A: _____ AVG TT: _____
- TT Job B: _____
- TT Job C: _____
- TT Job D: _____

- Calculate the response time for each job, and the average response time
- RT Job A: _____ AVG RT: _____
- RT Job B: _____
- RT Job C: _____
- RT Job D: _____

Q9 PART B – ROUND ROBIN SCHEDULER

- Using the graph, calculate the turnaround time for each job, and the average turnaround time
- TT Job A: $\underline{28-0=28}$ AVG TT: $\underline{100/4=25}$
- TT Job B: $\underline{31-2=29}$
- TT Job C: $\underline{33-10=23}$
- TT Job D: $\underline{36-16=20}$
- Calculate the response time for each job, and the average response time
- RT Job A: $\underline{\quad 0 \quad}$ AVG RT: $\underline{11/3=3.66}$
- RT Job B: $\underline{3-2 = 1}$
- RT Job C: $\underline{12-10=2}$
- RT Job D: $\underline{24-16=8}$

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.23

MULTI-LEVEL FEEDBACK QUEUE

- Review the bonus lecture for scheduling examples including several Multi-level-feedback-queue problems (MLFQ)
- Shortcut to Zoom recording of practice session:
<https://tinyurl.com/422w26-practice>

February 10, 2026

TCSS422: Operating Systems [Winter 2026]
School of Engineering and Technology, UW Tacoma

L10.24

QUESTIONS

