Exercise 1

A computer system has enough room to hold five programs in its main memory. These programs are idle waiting for I/O half the time. What fraction of the CPU time is wasted?

Exercise 2

A computer has 4 GiB of RAM of which the operating system occupies 512 MiB. The processes are all 256 MiB (for simplicity) and have the same characteristics. If the goal is 99% CPU utilization, what is the maximum I/O wait that can be tolerated?

Exercise 3

A real-time system needs to handle two voice calls that each run every 6 msec and consume 1 msec of CPU time per burst, plus one video at 25 frames/sec, with each frame requiring 20 msec of CPU time. Is this system schedulable?

Exercise 4

A real-time system has four periodic events with periods of 50, 100, 200, and 250 msec each. Suppose that the four events require 35, 20, 10, and x msec of CPU time, respectively. What is the largest value of x for which the system is schedulable?

Exercise 5

There are 8 batch jobs denoted with $\Pi_1, \Pi_2, \ldots, \Pi_8$. (Those jobs have arrived almost simultaneously, where those job with smaller numbers arrived slightly earlier.) Estimated running times are (in seconds):

Job	Π_1	Π_2	Π_3	Π_4	Π_5	Π_6	Π_7	Π_8
Running time	9	13	5	12	7	10	20	4

Write down the order of execution and determine the average response time using the following algorithms:

(a) first-come first-served,

(b) shortest job first.

Exercise 6

Five jobs are waiting to be run. Their expected run times are

$$9,\ 6,\ 3,\ 5,\ \text{and } X.$$

In what order should they be run to minimize average response time? (Your answer will depend on X.)

Exercise 7

There are 5 processes $\Pi_1, \Pi_2, \ldots, \Pi_5$:

(a) Processes $\Pi_1, \ldots, \Pi_4$ arrive in the given order at time 4 ms. Each of those processes needs 4 ms of CPU time.

(b) Process Π_5 arrives at time 20 ms and needs 6 ms of CPU time.

Scheduling is preemptive and the process switch takes 1 ms. For the *Round-Robin* algorithm with the quantum of 2 ms depict the scheduling of those processes on a timeline. Then determine the average turnaround time.

Exercise 8

Measurements of a certain system have shown that the average process runs for a time T before blocking on I/O. A process switch requires a time S, which is effectively wasted (overhead). For round-robin scheduling with quantum Q, give a formula for the CPU efficiency for each of the following:
(a) $Q = \infty$
(b) $Q > T$
(c) $S < Q < T$
(d) $Q = S$
(e) Q nearly 0

Exercise 9

The aging algorithm with $a = \frac{1}{2}$ is being used to predict run times. The previous four runs, from oldest to most recent, are 40, 20, 40, and 15 msec. What is the prediction of the next time?

Exercise 10

There are 5 periodic processes Π_1, Π_2, Π_3, Π_4 and Π_5 that are processing video. Their periods P_i and processing times C_i (in milliseconds) are given in the table below. For:

(a) the Rate Monotonic Scheduling (RMS) algorithm and

(b) the Earliest Deadline First (EDF) algorithm

draw a diagram that will depicts the scheduling of those processes for the first 150 ms.

Proces	P_i	C_i
Π_1	40	5
Π_2	60	10
Π_3	55	15
Π_4	35	5
Π_5	60	15

In case of a tie the process that is already running has priority. If no process is already running then use the one with the smallest consecutive number. Does any of the above algorithms fail?

Exercise 11

In this problem you are to compare reading a file using a single-threaded file server and a multithreaded server. It takes 12 msec to get a request for work, dispatch it, and do the rest of the necessary processing, assuming that the data needed are in the block cache. If a disk operation is needed, as is the case one-third of the time, an additional 75 msec is required, during which time the thread sleeps. How many requests/sec can the server handle if it is single threaded? If it is multithreaded?

Exercise 12

In the dining philosophers problem, let the following protocol be used:

(i) an even-numbered philosopher always picks up his left fork before picking up his right fork;

(ii) an odd-numbered philosopher always picks up his right fork before picking up his left fork.

Will this protocol guarantee deadlock-free operation?