THE UNIVERSITY of EDINBURGH
informatics

Operating Systems (INFR09047)

Revision 1

February 2020

Credit

TA Team

Dimitra Giantsidi
D.Giantsidi <at> sms.ed.ac.uk

Resul Tugay
resul.tugay <at> ed.ac.uk

Chris Vasiladiotis
c.vasiladiotis <at> sms.ed.ac.uk

THE UNIVERSITY of EDINBURGH
informatics

Practice Exercises from "Operating Systems Concepts" Silberschatz et al., 10th ed.

2.1, 2.5

3.1, 3.2, 3.5

4.1, 4.4

5.3, 5.4, 5.5

THE UNIVERSITY of EDINBURGH
informatics

Chapter 2

Practice exercises 2.1, 2.5

Practice Exercise 2.1

What is the purpose of system calls?

Practice Exercise 2.5

What is the main advantage of the layered approach to system design?

What are the disadvantages of the layered approach?

Practice Exercise 3.5

When a process creates a new process using the `fork()` operation, which of the following states is shared between the parent process and the child process?

a. Stack
b. Heap
c. Shared memory segments

Chapter 3

Practice exercises 3.1, 3.2, 3.5

Practice Exercise 3.1 (1/2)

Using the program shown in Figure 3.30, explain what the output will be at `LINE A`.

Practice Exercise 3.1 (2/2)

```
/* required includes truncated to save space */

int value = 5;

int main() {
      pid_t pid;
      pid = fork();

      if (pid == 0) { /* child process */
            value += 15;
            return 0;
      }
      else if (pid > 0) { /* parent process */
            wait(NULL);
            printf("PARENT: value = %d",value); /* LINE A */
            return 0;
      }
}
```

Figure 3.30

Practice Exercise 3.2 (2/2)

```
/* required includes truncated to save space */

int main()
{
      /* fork a child process */
      fork();

      /* fork another child process */
      fork();

      /* and fork another */
      fork();

      return 0;
}
```

Figure 3.31

THE UNIVERSITY *of* EDINBURGH
informatics

Practice Exercise 3.2 (1/2)

Including the initial parent process, how many processes are created by the program shown in Figure 3.31?

Practice Exercise 3.5

When a process creates a new process using the fork() operation, which of the following states is shared between the parent process and the child process?

a. Stack
b. Heap
c. Shared memory segments

Chapter 4

Practice exercises 4.1, 4.4

Practice Exercise 4.1

Provide three programming examples in which multithreading provides better performance than a single-threaded solution.

Practice Exercise 4.4

What are two differences between user-level threads and kernel-level threads?

Under what circumstances is one type better than the other?

Chapter 5

Practice exercises 5.3, 5.4, 5.5

Practice Exercise 5.3 (1/5)

Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed.

In answering the questions, use **non-preemptive** scheduling, and base all decisions on the information you have at the time the decision must be made.

Practice Exercise 5.3 (2/5)

a. What is the average turnaround time for these processes with the FCFS scheduling algorithm?

Process	Arrival Time	Burst Time
P1	0.0	8
P2	0.4	4
P3	1.0	1

Practice Exercise 5.3 (3/5)

b. What is the average turnaround time for these processes with the SJF scheduling algorithm?

Process	Arrival Time	Burst Time
P1	0.0	8
P2	0.4	4
P3	1.0	1

Practice Exercise 5.3 (4/5)

c. The SJF algorithm is supposed to improve performance, but notice that we chose to run process P1 at time 0 because we did not know that two shorter processes would arrive soon. Compute what the average turnaround time will be if the CPU is left idle for the first 1 unit and then SJF scheduling is used. Remember that processes P1 and P2 are waiting during this idle time, so their waiting time may increase. This algorithm could be known as ***future-knowledge scheduling***.

Practice Exercise 5.3 (5/5)

Process	Arrival Time	Burst Time
P1	0.0	8
P2	0.4	4
P3	1.0	1

Practice Exercise 5.4 (1/5)

Consider the following set of processes, with the length of the CPU burst time given in milliseconds *(see next slide)*.

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

Practice Exercise 5.4 (2/5)

a. Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, non-preemptive priority (a larger priority number implies a higher priority), and RR (quantum = 2).

Process	Burst Time	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

Practice Exercise 5.4 (3/5)

b. What is the turnaround time of each process for each of the scheduling algorithms in part a?

Process	Burst Time	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

Practice Exercise 5.4 (4/5)

c. What is the waiting time of each process for each of these scheduling algorithms?

Process	Burst Time	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

Practice Exercise 5.4 (5/5)

d. Which of the algorithms results in the minimum average waiting time (over all processes)?

Process	Burst Time	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

Practice Exercise 5.5 (1/5)

The following processes are being scheduled using a preemptive, round-robin scheduling algorithm. Each process is assigned a numerical priority, with a higher number indicating a higher relative priority.

In addition to the processes listed below, the system also has an idle task (which consumes no CPU resources and is identified as Pidle). This task has priority 0 and is scheduled whenever the system has no other available processes to run.

The length of a time quantum is 10 units. If a process is preempted by a higher-priority process, the preempted process is placed at the end of the queue.

Practice Exercise 5.5 (2/5)

a. Show the scheduling order of the processes using a Gantt chart.

Process	Priority	Burst	Arrival
P1	40	20	0
P2	30	25	25
P3	30	25	30
P4	35	15	60
P5	5	10	100
P6	10	10	105

Practice Exercise 5.5 (3/5)

b. What is the turnaround time for each process?

Process	Priority	Burst	Arrival
P1	40	20	0
P2	30	25	25
P3	30	25	30
P4	35	15	60
P5	5	10	100
P6	10	10	105

Practice Exercise 5.5 (4/5)

c. What is the waiting time for each process?

Process	Priority	Burst	Arrival
P1	40	20	0
P2	30	25	25
P3	30	25	30
P4	35	15	60
P5	5	10	100
P6	10	10	105

THE UNIVERSITY of EDINBURGH informatics

Practice Exercise 5.5 (5/5)

d. What is the CPU utilization rate?

Process	Priority	Burst	Arrival
P1	40	20	0
P2	30	25	25
P3	30	25	30
P4	35	15	60
P5	5	10	100
P6	10	10	105

The End.

Thank you!