

Exercise walk-through

1.4

Operating systems

International Baccalaureate · IB Diploma · Computer Science · HL

Computing Systems · executable trace

- Trace the state changes and boundary guard.
- Run the boundary case and compare the printed output.

```
size_MB, rate_Mbps = 20, 10
bits = size_MB * 1_000_000 * 8
if rate_Mbps <= 0:
    raise ValueError("Rate must be positive")
seconds = bits / (rate_Mbps * 1_000_000)
print(seconds)
print(0 / (rate_Mbps * 1_000_000))
```

Exercises 2.1-2.3

2.1 State **three** functions of an operating system.

[3]

2.2 Processes A (3 ms), B (5 ms) and C (2 ms) arrive at 0 in that order. State when each ends under (i) first come first served; (ii) round robin with a 2 ms slice.

[4]

2.3 Explain why the CPU saves its registers before running an interrupt handler.

[2]

Exercises 2.1-2.3

2.1 State **three** functions of an operating system. [3]

2.1

- any three, e.g. process, memory, file or device management, security, user interface

2.2 Processes A (3 ms), B (5 ms) and C (2 ms) arrive at 0 in that order. State when each ends under (i) first come first served; (ii) round robin with a 2 ms slice. [4]

2.2

- (i) A at **3**, B at **8**, C at **10**
- (ii) A 0–2, B 2–4, C 4–6 (ends), A 6–7 (ends), B 7–10 (ends): C **6**, A **7**, B **10**

2.3 Explain why the CPU saves its registers before running an interrupt handler. [2]

2.3

- the handler uses the registers too, so their values would be lost
- saving them lets the interrupted program carry on exactly where it stopped

Exercises 3.1-3.2

3.1 Which OS function stops one program overwriting another's data in RAM?

[1]

☐

A file management

B memory management

C device management

D the user interface

3.2 What is a weakness of priority scheduling?

[1]

☐

A short jobs always wait

B low-priority processes may never run

C every process gets the same time

D it cannot run on one core

Exercises 3.1-3.2

3.1 Which OS function stops one program overwriting another's data in RAM?

[1]

☐

A file management

B memory management

C device management

D the user interface

3.1 B

- each program has its own protected area

3.2 What is a weakness of priority scheduling?

[1]

☐

A short jobs always wait

B low-priority processes may never run

C every process gets the same time

D it cannot run on one core

3.2 B

- starvation

Exercise 3.3

3.3 Which event is best handled by an interrupt?

[1]

☐

A reading a temperature sensor every second

B a printer running out of paper

C updating a clock display every minute

D copying a file

Exercise 3.3

3.3 Which event is best handled by an interrupt?

[1]

☐

A reading a temperature sensor every second

B a printer running out of paper

C updating a clock display every minute

D copying a file

3.3 B

- rare and unpredictable

Exercise 3.4 (a)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(a) Explain how the operating system makes the three apps appear to run at the same time. [2]

Exercise 3.4 (a)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(a) Explain how the operating system makes the three apps appear to run at the same time. [2]

(a)

- the scheduler gives each app a short time slice of the CPU in turn
- it switches so fast that all three seem to run together

Exercise 3.4 (b)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(b) The game needs 6 ms, the music app 2 ms and the messaging app 3 ms. Using round robin with a 2 ms slice, in that order, state when each finishes. [3]

Exercise 3.4 (b)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(b) The game needs 6 ms, the music app 2 ms and the messaging app 3 ms. Using round robin with a 2 ms slice, in that order, state when each finishes. [3]

(b)

- game 0–2, music 2–4 (ends), messaging 4–6, game 6–8, messaging 8–9 (ends), game 9–11 (ends)
- music **4 ms**, messaging **9 ms**
- game **11 ms**

Exercise 3.4 (c)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(c) A message arrives while the game is running. Describe how the interrupt is handled. [3]

Exercise 3.4 (c)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(c) A message arrives while the game is running. Describe how the interrupt is handled. [3]

(c)

- the CPU finishes its current instruction and saves its registers
- it runs the interrupt handler for the message
- it restores the registers and the game carries on

Exercise 3.4 (d)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(d) Explain why the music app should be given a higher priority than the game. [2]

Exercise 3.4 (d)

3.4 A smartphone runs a music app, a messaging app and a game at the same time on one core.

(d) Explain why the music app should be given a higher priority than the game. [2]

(d)

- music must play without gaps: a late slice is heard as a glitch
- a slightly slower game frame is less noticeable