

THEME A: CONCEPTS · A1.3

Operating systems and control systems — Mark scheme

34 marks · covers A1.3.1–A1.3.7

How to mark. Each creditworthy point is on its own line and ends with a semicolon (;). A line beginning with ... is an *expansion* of the line above it, so the pair is worth the marks shown. A slash (/) separates acceptable alternative wordings. Words in (brackets) are not needed for the mark. Mark positively: credit what the student has got right rather than penalising what is missing, and accept any valid alternative not listed here. Do not award more than the maximum for a part. Where a later part uses a wrong answer from an earlier part correctly, award follow-through marks (FT). HL marks parts that assess higher level only content.

Q	ANSWER	MARKS
1(a) A1.3.1	Award [2] max Award [1] per point up to [2] max. The operating system provides an abstraction layer / a simple interface between the application and the hardware; The application asks the operating system to save a file or draw to the screen, rather than controlling the device directly; Device drivers deal with the differences between individual pieces of hardware, so the same program works on different machines; The user works through a graphical user interface (GUI) and does not need to know how the hardware operates;	2
1(b) A1.3.2	Award [3] max Award [1] per point up to [3] max. The operating system allocates a separate area of main memory to each program; It keeps a record of which areas are in use and which are free, and reclaims memory when a program closes; It prevents one program from reading or writing the memory allocated to another, so the browser cannot corrupt the photo editor; If the two programs together need more memory than is physically available, less-used pages are moved to secondary storage (virtual memory); This allows both programs to be held in memory at the same time so the CPU can switch between them;	3
2(a) A1.3.2	Award [2] max Award [1] for the point and [1] for the expansion. The answer must be linked to identifying heavy users. The accounting function logs how much of each resource (CPU time, memory, disk, network) every user or process consumes; ... (so) these logs can be totalled per student to show who has used the most processor time; Usage is tracked against each individual login/account; ... (so) the university can rank users, charge them, or cap the share of the server they are allowed;	2
2(b) A1.3.2	Award [2] max	2

Q	ANSWER	MARKS
	Award [1] for the point and [1] for the expansion. Any one reason. Virtualization lets several isolated virtual machines run on the one physical server; ...(so) each student or course can have its own environment without the university buying more hardware; Each virtual machine is isolated from the others; ...(so) a crash or a security breach in one does not affect the others; Virtual machines can be created, copied and deleted quickly; ...(so) a standard environment can be provided to every student and reset easily between classes;	
2(c) A1.3.2	Award [1] max Any one, [1] max. Authenticating users with usernames and passwords; Controlling access permissions to files and directories; Encrypting stored data; Maintaining an audit log of access; Isolating processes so one user cannot read another user's memory;	1
3(a) A1.3.3	Award [3] max Award [1] per point up to [3] max. The answer must refer to the processes in Table 1. First-come first-served is non-pre-emptive, so each process runs to completion before the next one starts; P2 arrives at the same time as the others but sits behind P1 in the queue; P1 needs 6 ms, so P2 cannot start until 6 ms have passed even though it only needs 2 ms of CPU time; A short process is therefore held up behind a longer one, which is known as the convoy effect; This gives P2 a poor response time relative to the work it actually has to do;	3
3(b) A1.3.3	Award [2] max Award [1] per point up to [2] max. Must be applied to the four processes. Round robin is pre-emptive: each process is given a fixed time slice (quantum) in turn; P1 would be interrupted after its quantum and moved to the back of the queue, so P2 starts far sooner; All four processes would make progress from early on rather than finishing strictly one after another; The short process P2 would finish much earlier, though the longest process P3 would finish later than under FCFS; Switching between processes adds context-switching overhead;	2
3(c) A1.3.3	Award [1] max Multilevel queue scheduling;	1
3(d) A1.3.3	Award [1] max Any one, [1] max. Low-priority processes may never get the CPU (starvation), unless ageing is used to raise their priority over time; Priorities have to be assigned correctly, and a badly chosen priority damages performance;	1

Q	ANSWER	MARKS
	It can be unfair, because a process is judged on its priority rather than on how long it has waited;	
4(a) A1.3.4	Award [2] max Award [1] per point up to [2] max. Must describe polling, not interrupts. The processor would check the state of the button at fixed, regular intervals; It reads the button's status each time round the loop and compares it with the previous reading; If the button has been pressed the processor runs the code that responds; if not, it carries on and checks again next time; The press is therefore not noticed until the next scheduled check, so the response is delayed by up to one polling interval; Do not credit an answer that describes the device signalling the processor, which is interrupt handling.	2
4(b) A1.3.4	Award [4] max This is an evaluate question (AO3), so both strengths and limitations are required. Award [1] per point up to [4] max, but award a maximum of [3] if only one side is given. Points must be linked to this device. Strengths of interrupts here A button press is unpredictable and rare, so polling would spend almost all its checks finding nothing; The CPU can stay in a low-power sleep state until the interrupt arrives, which matters because the tracker runs on a battery; The response is immediate rather than waiting until the next scheduled poll, so the device feels responsive to the user; Limitations of interrupts here Each interrupt carries an overhead of saving and restoring the processor state, which costs time and energy; A button that bounces could generate many interrupts in quick succession, needing extra handling in software or hardware; Interrupt-driven code is harder to write and test than a simple polling loop; Polling remains the better choice for the heart-rate sensor, which is read at known regular intervals; Judgement (credit a supported conclusion, for example) Interrupts are the better choice for the button because the event is rare and unpredictable and battery life is the priority, while polling suits the predictable sensor readings;	4
HL 5(a) A1.3.5	Award [2] max Award [1] for the point and [1] for the expansion. Resource contention occurs when two or more processes need the same resource at the same time and it cannot be shared; ... (so) one process must wait while the other uses it, which slows the waiting process down; Several processes compete for a limited resource such as the printer, the disk or a region of memory; ... (so) the operating system must decide which process gets it and in what order;	2
HL 5(b) A1.3.5	Award [3] max Award [1] per point up to [3] max. A worked illustration using the printer and the disk is acceptable. Each process holds one resource while requesting the other;	3

Q	ANSWER	MARKS
	Process A holds the printer and asks for the disk, while process B holds the disk and asks for the printer; Neither will release the resource it already holds until it gets the one it has asked for; Neither request can ever be granted, so both processes wait forever and no progress is made; The operating system must detect this and break the cycle, for example by forcing one process to release its resource;	
HL 6(a) A1.3.6	Award [2] max Award [1] for each. Sensor: the temperature sensor / thermistor (which measures the air temperature); Actuator: the motor (which opens and closes the roof vent);	2
HL 6(b) A1.3.6	Award [3] max Award [1] per point up to [3] max. The output of the system (the resulting air temperature) is measured by the sensor and fed back into the controller as an input; The controller compares this measured value with the set value chosen by the gardener; The size and direction of the difference decides the next action, so the vent is opened further, held where it is, or closed; Because the loop repeats continuously, the temperature is pulled back towards the set value without the gardener intervening; It therefore corrects automatically for outside disturbances such as the sun coming out or the weather turning cold; Do not award marks for an open-loop description, in which the output is never measured and so cannot influence the next action.	3
HL 6(c) A1.3.7	Award [1] max Any one, [1] max. A home thermostat / central heating system; Cruise control in a car; An automatic washing machine; A lift/elevator controller; A traffic signal control system that responds to sensed traffic; An irrigation control system that responds to soil moisture; The device named must genuinely use feedback. Do not credit an open-loop example such as a microwave timer or a set of traffic lights on a fixed cycle.	1

Total for this paper: 34 marks