



F9.9 Operating systems and utilities

Logic and computer systems · GCSE · OCR J277 1.5.1, AQA 8525 3.4.3,
Edexcel 1CP2 3.2.1 · about 15 min

Name _____

Class _____

Date _____

What this lesson is about

What an operating system does, scheduling, and utility software.

- What an operating system does
- Sharing the processor
- Utility software

Questions

6 marks in all

- What is a device driver? [1 mark]
 - ☐ A A program that lets the operating system talk to a particular device
 - ☐ B A person who fixes hardware
 - ☐ C A utility that speeds up the disk
 - ☐ D The part of the CPU that controls devices
- How does an operating system run several programs on one core? [1 mark]
 - ☐ A It gives each a short time slice and switches between them quickly
 - ☐ B It runs them all at the same instant
 - ☐ C It runs one program to the end before starting the next
 - ☐ D It copies each program to a separate core
- What does defragmentation do? [1 mark]
 - ☐ A Moves the pieces of each file on a hard disk back together
 - ☐ B Removes viruses
 - ☐ C Makes files smaller
 - ☐ D Copies files to another drive
- Why should an SSD not be defragmented? [1 mark]
 - ☐ A It gains no speed and the extra writes wear it out
 - ☐ B It deletes the files
 - ☐ C SSDs cannot store files in pieces
 - ☐ D It makes the SSD volatile
- What does an incremental backup copy? [1 mark]
 - ☐ A Only the files changed since the last backup
 - ☐ B Every file on the computer
 - ☐ C Only the operating system
 - ☐ D Only files that have been deleted

6. What does this program print?

[1 mark]

```
queue = ["a", "b", "c"]
job = queue.pop(0)
queue.append(job)
print(queue)
```

The task: a round-robin scheduler

Write a round-robin scheduler with a time slice of 2 units. Each job in `jobs` needs some units of processor time. Take the job at the front of the queue, run it for 2 units or for what it has left, whichever is less, and print `<name> ran for <n>`. If it is not finished, put it at the back of the queue; if it is, print `<name> finished at <t>`, where `t` is the total time so far.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

jobs = [("motors", 3), ("camera", 5), ("radio", 2)]
time_slice = 2
```

Plan your program here, then type it in and press Run.



Do it on the robot

www.bugbotlab.com/learn/f9-9-operating-systems-and-utilities/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. Try a time slice of 1 and of 10. Which is fairest? Which switches least?
2. Give each job a priority and always run the highest priority job that is waiting.
3. Make an incremental backup: given yesterday's files and today's, print only the files that are new or changed.