



A10.1 Classifying software

Operating systems, software and translators · A level · OCR H446 1.2.2,
AQA 7517 4.6.1.2, Eduqas A500QS 2.7 · about 30 min

What this lesson is about

System and application software, utilities, libraries and translators, and open versus closed source.

- Two classes of software
- System software
- Application software
- Open source and closed source

Questions

5 marks in all

1. Which of these are system software?

[1 mark]

Tick every answer that is true.

- ☐ A A compiler
- ☐ B A disk defragmenter
- ☐ C A maths library
- ☐ D A spreadsheet package
- ☐ E A payroll program

Answer: A, B, C. Translators, utilities and libraries all exist to run or support the computer and other programs; a spreadsheet and payroll do a job for the user.

2. What is a library?

[1 mark]

- ☐ A A collection of pre-written, tested routines that other programs can call
- ☐ B A program that translates source code into machine code
- ☐ C A folder where the operating system keeps its files
- ☐ D An application for searching documents

Answer: A. A library has no main program of its own; programs link to its subroutines.

3. A small charity needs software to record donations. Which is the strongest reason to choose off-the-shelf software over bespoke?

[1 mark]

- ☐ A It is available now, costs less, and has been tested by many users
- ☐ B It will match the charity's exact needs
- ☐ C The charity will own the source code
- ☐ D It never needs updates

Answer: A. Bespoke software fits exactly and is owned by the client, but it is expensive and slow to produce; a small charity benefits from low cost and immediate availability.

4. What makes software open source?

[1 mark]

- ☐ A Its source code is available, under a licence that lets people study, change and share it
- ☐ B It is free to download
- ☐ C It was written by volunteers
- ☐ D It has no licence at all

Answer: A. Free of charge is not the same thing: closed source software can be free, and open source software is still licensed.

5. Which is a disadvantage of closed source software for the organisation that buys it?

[1 mark]

- ☐ A Only the owner can find and fix flaws or add features
- ☐ B Attackers can read its source code
- ☐ C There is never any official support
- ☐ D It cannot be sold

Answer: A. The buyer cannot change the code, so it depends on the company for fixes and changes.

The task: the software audit

Audit the software on a robot and the laptop that programs it. `software` is a list of tuples `(name, kind, licence)`: `name` and `kind` are strings, and `licence` is "open" or "closed". `kind` is one of "operating system", "device driver", "translator", "utility", "library" or "application". Write `classify(kind)`, which returns the string "system" for the five kinds of system software and "application" for applications. Then, for each item in order, print `<name>: <group>, <licence> source`, for example `FreeRTOS: system, open source`. After the list, print three totals, one per line: `system: <n>`, `application: <n>` and `open source: <n>`. Count them in your loop rather than typing them. The robot stays still.

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

software = [
    ("FreeRTOS", "operating system", "open"),
    ("motor driver", "device driver", "open"),
    ("pocketpy", "translator", "open"),
    ("backup tool", "utility", "closed"),
    ("lesson browser", "application", "closed"),
    ("maths library", "library", "open"),
    ("spreadsheet", "application", "closed"),
    ("disk defragmenter", "utility", "closed"),
]

def classify(kind):
    pass
```

The hint students can ask for: Decide which kinds of software exist to run and look after the computer, and which exist to do a job for the user. Keep running counts in the same loop that prints each line, and print the totals after the loop.

A solution

```
from bugbot import *
connect()

software = [
    ("FreeRTOS", "operating system", "open"),
    ("motor driver", "device driver", "open"),
    ("pocketpy", "translator", "open"),
    ("backup tool", "utility", "closed"),
    ("lesson browser", "application", "closed"),
    ("maths library", "library", "open"),
    ("spreadsheet", "application", "closed"),
    ("disk defragmenter", "utility", "closed"),
]

SYSTEM_KINDS = ["operating system", "device driver", "translator", "utility", "library"]

def classify(kind):
    if kind in SYSTEM_KINDS:
        return "system"
    return "application"

counts = {"system": 0, "application": 0}
```

```
open_count = 0
for name, kind, licence in software:
    group = classify(kind)
    counts[group] = counts[group] + 1
    if licence == "open":
        open_count = open_count + 1
    print(f"{name}: {group}, {licence} source")
print("system:", counts["system"])
print("application:", counts["application"])
print("open source:", open_count)
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.