

2.1 COMPUTATIONAL THINKING · 2.1.3

Thinking procedurally

Original practice questions · 30 marks · about 40 minutes · spec 2.1.3(a)–(d)

Instructions. Answer all questions. The number of marks is shown in brackets []. Name sub-procedures like routines, e.g. checkStock.

1 Total: 6 marks

A developer is writing a quiz game app that asks questions, scores answers and shows a final result.

(a) State what is meant by *decomposition*. [1]

(b) Identify **three** sub-procedures the quiz game could be decomposed into. [3]

(c) Give one benefit to the developer of decomposing the problem this way. [2]

2 Total: 6 marks

A structure diagram for a cinema booking system has processBooking at the top, with sub-procedures selectFilm and chooseSeats shown beneath it.

(a) Identify **two** further sub-procedures needed to complete the structure diagram. [2]

(b) State what a structure diagram shows. [2]

(c) State what is meant by a *sub-procedure*. [2]

3

Total: 6 marks

A library self-service machine uses these sub-procedures: `scanCard`, `scanBook`, `checkLimit`, `recordLoan`, `printReceipt`.

(a) State the correct order in which these sub-procedures should run. [2]

(b) Explain why `checkLimit` must run before `recordLoan`. [2]

(c) Identify a sub-procedure above that could be decomposed further, and give **two** smaller sub-procedures it could contain. [2]

4

Total: 6 marks

A vending machine dispenses a drink after the user selects an item and inserts enough money.

(a) Identify **three** components of this problem. [3]

(b) Identify **two** sub-procedures of the solution. [2]

(c) Name one sub-procedure that must run before the drink is dispensed. [1]

5

Total: 6 marks

(a) Explain the difference between *decomposition* and *abstraction*. [2]

(b) Give **two** benefits of using sub-procedures in a solution. [2]

(c) Explain why determining the order of the steps is part of thinking procedurally. [2]

END OF QUESTIONS