

2.2 PROGRAMMING TECHNIQUES · 2.2.2(D)–(F)

Divide & conquer, backtracking & heuristics

Original practice questions · 30 marks · about 40 minutes · spec 2.2.2(d)–(f)

Instructions. Answer all questions. Marks are shown in brackets []. Give an example where asked.**1** Total: 6 marks**(a)** State what is meant by *divide and conquer* and name one example algorithm. [2]

(b) State what is meant by a *heuristic*. [2]

(c) State what is meant by *backtracking*. [2]

2 Total: 6 marks**(a)** Explain why a quicksort is described as a divide-and-conquer algorithm. [2]

(b) State the precondition that must hold before a binary search can be used. [1]

(c) Explain how a binary search is an example of divide and conquer. [3]

3 Total: 6 marks**(a)** Name the computational method best described by each scenario: [3]

- (i) A robot escapes a maze, reversing whenever it reaches a dead end.
- (ii) A supermarket discovers which products are frequently bought together.
- (iii) A delivery app quickly finds a near-optimal route without checking every option.

(b) Describe what is meant by *data mining* and give one example of its use. [3]

4

Total: 6 marks

(a) Explain when a heuristic approach would be used instead of an exact method. [2]

.....

(b) Explain why a heuristic may not produce the best possible solution. [2]

.....

(c) Give one real situation where a heuristic is appropriate, with a reason. [2]

.....

5

Total: 6 marks

(a) Describe what is meant by *pipelining*. [2]

.....

(b) State one difference between pipelining and parallel processing. [2]

.....

(c) State one benefit of visualising data. [1]

.....

(d) State what performance modelling is used for. [1]

.....

END OF QUESTIONS