1.3 Exchanging Data: Databases, Networks and Web Technologies: 10-minute Pack

OCR H446 · Calculators not allowed · about 10 minutes

Total Marks

Name: _____ Date: ____ / ____ / _____

Answer ALL questions. Show all your working.

1 A programmer is implementing a hash table of size 10 (indices 0 to 9) using the hash function h(k) = k MOD 10, with collisions resolved by linear probing (checking the next index upward, wrapping to 0 if necessary).

(a) Explain the purpose of a hash function when storing data in a hash table. **(2)**

(b) The keys 23, 7, 17 and 13 are inserted, in that order, into the empty hash table. Show the index at which each key is placed, and give the final contents of the whole table (indices 0 to 9). **(4)**

(c) State one drawback of using modulo division as a hash function. **(1)**

(Total for Question 1 is 7 marks)

© Revision Library | revisionlibrary.co.uk

Mark scheme · 1.3 Exchanging Data: Databases, Networks and Web Technologies: 10-minute Pack

Question 1

(a) States that the hash function converts a key into an index/address (oe) **M1**

(a) Explains this allows data to be stored and retrieved directly/in near constant time, without searching every record (oe) **A1**

(a) Answer: A hash function converts a key into a table index, allowing data to be stored at and retrieved from that index directly rather than being searched for.

(b) 23 -> index 3 and 7 -> index 7 both placed with no collision **M1**

(b) 17 correctly identified as colliding at index 7 and probed to index 8 **M1**

(b) 13 correctly identified as colliding at index 3 and probed to index 4 **M1**

(b) Fully correct final table cao: index 3 = 23, index 4 = 13, index 7 = 7, index 8 = 17, all other indices empty **A1**

(b) Answer: Index 0: empty, 1: empty, 2: empty, 3: 23, 4: 13, 5: empty, 6: empty, 7: 7, 8: 17, 9: empty

(c) Keys that share a common factor with the table size (or are multiples of it) cluster/collide at the same or nearby indices, especially if the table size is not prime (oe) **B1**

(c) Answer: Keys sharing a common factor with the table size can cluster at the same indices, causing more collisions than a well-chosen table size/hash function.

© Revision Library | revisionlibrary.co.uk