Started on	Sunday, 7 May 2023, 10:29 PM
State	Finished
Completed on	Sunday, 7 May 2023, 10:46 PM
Time taken	16 mins 26 secs
Grade	**10.00** out of 10.00 (**100**%)

Question 1

Correct

Mark 1.00 out of 1.00

In the context of Hash Tables, what does the term '**open addressing**' refer to?

- a. Storing collided elements in the same array as the Hash Table itself ✔
- b. Storing collided elements in an auxiliary data structure
- c. Storing all elements in a single linked list
- d. Storing collided elements in separate data structures

The correct answer is:
Storing collided elements in the same array as the Hash Table itself

Question 2

Correct

Mark 1.00 out of 1.00

Given the values {2341, 4234, 2839, 430, 22, 397, 3920}, a hash table of size 7, and hash function h(x) = x mod 7, select the resulting tables after inserting the values in the given order with linear probing.

- a. 0 [430] 1 [22] 2 [3920] 3 [2341] 4 [2839] 5 [397] 6 [4234]
- b. 0 [397] 1 [22] 2 [3920] 3 [2341] 4 [2839] 5 [430] 6 [4234] ✔
- c. 0 [3920], 1 [22], 2 [ ], 3 [2341, 430], 4 [2839], 5 [397], 6 [4234]
- d. 0 [3920] 1 [430] 2 [22] 3 [2341] 4 [2839] 5 [397] 6 [4234]

The correct answer is: 0 [397] 1 [22] 2 [3920] 3 [2341] 4 [2839] 5 [430] 6 [4234]

Question 3

Correct

Mark 1.00 out of 1.00

Which index will the character 'a' hash to in a **Hash Table** with $M = 7$, with $H(k) = k \% M$ (where we use key k's ASCII value in the mod operation), and where we are using 0-based indexing?
Hint- ASCII value of 'a' is 97

Answer: 6 ✔

The correct answer is: six

Question 4

Correct

Mark 1.00 out of 1.00

Given the following input (4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199) and the hash function x mod 10, which of the following statements are true?

i. 9679, 1989, 4199 hash to the same value

ii. 1471, 6171 has to the same value

iii. All elements hash to the same value

iv. Each element hashes to a different value

a. i only
b. ii only
c. i and ii only ✔
d. iii or iv

The correct answer is: i and ii only

Question 5

Correct

Mark 1.00 out of 1.00

How are collisions resolved in a hash table?

a. By storing all values that hash to the same index in a binary tree.
b. By overwriting the previous value with the new value.
c. By storing all values that hash to the same index in a linked list. ✔
d. By storing all values that hash to the same index in a hash table.

The correct answer is: By storing all values that hash to the same index in a linked list.

Question 6

Correct

Mark 1.00 out of 1.00

Consider following hash table of length 10, it uses addressing with hash function h(k)=k mod 10, and linear probing. After inserting 6 values table looks like following. What is the correct inserting order of numbers?

0	1	2	3	4	5	6	7	8	9
-	-	42	23	34	52	46	33	-	-

a. 46, 34, 42, 23, 52, 33 ✔

b. 46, 42, 34, 52, 23, 33

c. 42, 46, 33, 23, 34, 52

d. 34, 42, 23, 52, 33, 46

The correct answer is: 46, 34, 42, 23, 52, 33

Question 7

Correct

Mark 1.00 out of 1.00

Suppose you have a Hash Table that has 500 slots. It currently holds 99 keys, all in different locations in the Hash Table (i.e., there are no collisions thus far). What is the probability that the next key you insert will cause a collision? **Input your answer in decimal form and round to the nearest 3 digits.**

Answer: 0.198 ✔

The correct answer is: 0.198

Question 8

Correct

Mark 1.00 out of 1.00

Possible hash function to store string in hash table is the string's length, h(x) = x.length. This a good hash function.

Select one:

True

False ✔

Strings with the same length will have the same hash code.If we insert lots of strings with the same length, lookup will take O(n) time instead of O(1)

The correct answer is 'False'.

Question 9

Correct

Mark 1.00 out of 1.00

Disadvantages of Direct Address table

- [ ] a. One to one mapping
- [ ] b. Efficient operations
- [x] c. Have to consider huge universe of keys ✔
- [ ] d. Simple implementation

The correct answer is: Have to consider huge universe of keys

Question 10

Correct

Mark 1.00 out of 1.00

The time complexity of all operations associated with Direct Address table are not O(1)

Select one:

- ○ True
- ◉ False ✔

The correct answer is 'False'.