

Started on	Friday, September 29, 2023, 11:45 PM
State	Finished
Completed on	Friday, September 29, 2023, 11:53 PM
Time taken	7 mins 44 secs
Grade	10.00 out of 10.00 (100%)

Question 1

Correct

1.25 points out of 1.25

What is the benefit of a hash table?

- ☐ a. Uses less memory than a binary search tree.
- ☐ b. Fast $O(1)$ successor() and minimum() operations on average, like a tree.
- ☒ c. Fast $O(1)$ search and insert operations on average. ✓
- ☐ d. All of the above.
- ☐ e. None of the above.

Your answer is correct.

The correct answer is:

Fast $O(1)$ search and insert operations on average.**Question 2**

Correct

1.25 points out of 1.25

What is a weakness of direct addressing?

- ☒ a. It wastes a lot of space if the universe of keys is large in relation to the actual keys. ✓
- ☐ b. Search and insert take relatively slow $O(n)$ time.
- ☐ c. The hash function used must be carefully chosen.
- ☐ d. All of the above.
- ☐ e. None of the above.

Your answer is correct.

The correct answer is:

It wastes a lot of space if the universe of keys is large in relation to the actual keys.

Question 3

Correct

1.25 points out of 1.25

If there are 19 available slots, and the hash function is $k \bmod 19$, into which slot does 37 hash?

- ☐ a. 16
- ☐ b. 17
- ☒ c. 18 ✓
- ☐ d. 19
- ☐ e. 20

Your answer is correct.

The correct answer is:

18

Question 4

Correct

1.25 points out of 1.25

When two or more keys hash to the same slot, is called:

- ☐ a. Revision
- ☒ b. Collision ✓
- ☐ c. Decision
- ☐ d. Arch-Revival
- ☐ e. None of the Above

Your answer is correct.

The correct answer is:

Collision

Question 5

Correct

1.25 points out of 1.25

When chaining, what is the benefit of inserting a new item at the beginning of the linked list?

- ☒ a. Insert is $O(1)$ if you assume the item isn't already there, because you don't have to traverse the list to the end. ✓
- ☐ b. Search is $O(1)$ because the searched-for item will almost always be at the front of the list.
- ☐ c. Delete is $O(1)$ because the deleted item will almost always be at the front of the list.
- ☐ d. All of the above are benefits.
- ☐ e. None of the above are benefits.

Your answer is correct.

The correct answer is:

Insert is $O(1)$ if you assume the item isn't already there, because you don't have to traverse the list to the end.

Question 6

Correct

1.25 points out of 1.25

What is the worst case that can happen with chaining?

- ☐ a. All items hash to the same slot, which leads to an $O(\lg n)$ search.
- ☒ b. All items hash to the same slot, which leads to an $O(n)$ search. ✓
- ☐ c. Each item hashes to its own slot, leading to an $O(1)$ search.
- ☐ d. All of the above happen in the worst case.
- ☐ e. None of the above represent the worst case.

Your answer is correct.

The correct answer is:

All items hash to the same slot, which leads to an $O(n)$ search.

Question 7

Correct

1.25 points out of 1.25

What is a good choice for a hash function?

- ☐ a. $H(k) = k \bmod m$, where k is the number of keys, and $m = 64$ is the number of slots.
- ☒ b. $H(k) = k \bmod m$, where k is the number of keys and m is the number of slots. M is also chosen such that it is a prime number not close to a power of 2. ✓
- ☐ c. $H(k) = k / m$, where m is a prime number.
- ☐ d. All of the above are good hash functions.
- ☐ e. None of the above are good hash functions.

Your answer is correct.

The correct answer is:

$H(k) = k \bmod m$, where k is the number of keys and m is the number of slots. M is also chosen such that it is a prime number not close to a power of 2.

Question 8

Correct

1.25 points out of 1.25

How are keys stored in open addressing?

- ☐ a. In a linked list.
- ☐ b. In a two-dimensional array.
- ☒ c. In the hash table itself. ✓
- ☐ d. In an exterior file that is easy to search.
- ☐ e. In a searchable binary tree.

Your answer is correct.

The correct answer is:

In the hash table itself.