Summary in Graph

Exam Summary (GO Classes Test Series 2024 | Data Structures | Test 5)

| Qs. Attempted: | 13 ₄₊₉ | Correct Marks: | 16 4 + 12 | |
|---------------------|--------------------------|------------------|------------------|--|
| Correct Attempts: | 10 ₄₊₆ | Penalty Marks: | 0.67 0+0.67 | |
| Incorrect Attempts: | 3 | Resultant Marks: | 15.33 | |

EXAM STATS

FEEDBACK

EXAM RESPONSE

Technical

Q #1 Multiple Choice Type Award: 1 Penalty: 0.33 Algorithms

Suppose we use a hash function h to hash n distinct keys into an array T of length m. Say that two distinct keys x,y collide under h if h(x)=h(y). Assuming simple uniform hashing - that is, with each key mapped independently and uniformly to a random bucket - what is the probability that a given pair x,y of distinct keys collide?

A.
$$\frac{1}{m-1}$$
B. $\frac{1}{m}$
C. $\frac{1}{n^2}$
D. $\frac{1}{m^2}$

Your Answer: B Correct Answer: B Correct Discuss

Q #2 Multiple Select Type Award: 1 Penalty: 0 Algorithms

Let we insert the values 74,924,83,113, and 5 in the given order into a hash table of size 10, using the hash function k%10 and resolving collisions with quadratic probing. Determine which items we would encounter in order to search for the value 65?

- A. 924
- B. 5
- C. 83
- D. 113

Your Answer: A;B

Correct Answer: A;B

Correct **Discuss**

Penalty: 0.33

Q #3

Multiple Choice Type

Award: 1

Algorithms

Suppose M randomly selected keys are hashed into the range $[0 \dots N-1]$ using a uniformly distributed hashing function. What is the probability that all M keys yield the same hash value (so all M keys collide at the same slot)?

- D. None of these

Your Answer: A

Correct Answer: A

Correct

Discuss

Q #4

Multiple Choice Type

Award: 1

Penalty: 0.33

Algorithms

Suppose k randomly selected keys are hashed into the range $[0 \dots n-1]$ using a uniformly distributed hashing function. Assume that $k \leq n$. What is the probability that all k keys yield distinct hash values (so no collisions occur)?

- A. $\frac{(n-1)!}{(n-k)!n^{k-1}}$

- B. $\frac{(n-1)!}{(n-(k-1))!n^k}$ C. $\frac{n!(k-1)}{(n-(k-1))!n^k}$
- D. None of these

Your Answer: A

Correct Answer: A

Correct

Discuss

Q #5

Numerical Type

Award: 1

Penalty: 0

Algorithms

Assuming the simple uniform hashing is implemented into a hash table with m=3 slots. What is the probability for a key to map to slot 1 given that key does not map to slot 0?

Your Answer:

Correct Answer: 0.5

Not Attempted

Discuss

Q #6

Numerical Type

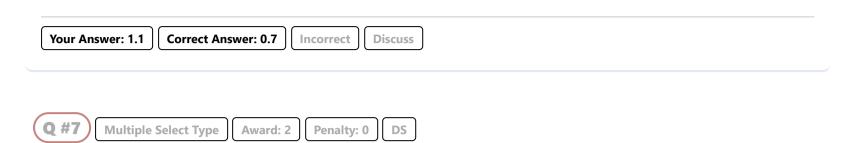
Award: 2

Penalty: 0

Algorithms

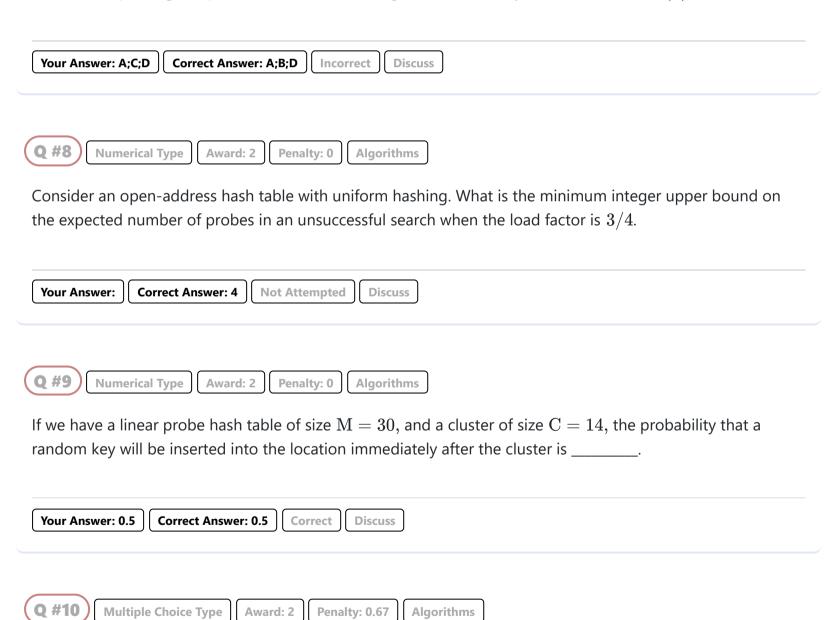
Given a hashing table of 10 entries, the hash function $h(x) = x \mod (10)$ and the input sequence $\{71, 23, 73, 99, 44, 79, 89\}$

What will be the load factor of the hash table when collisions are resolved with open addressing and linear probing?



Which of the following is is/are true?

- A. After hashing n keys into a hash table of size m that uses chaining to handle collisions, we hash two new keys k_1 and k_2 . Under the simple uniform hashing assumption, the probability that k_1 and k_2 are hashed into the same table location is exactly 1/m with no dependence on the number of keys n.
- B. Under the uniform hashing assumption, if we use a hash table of size m with open addressing to hash 3 keys, the probability that the third inserted key needs exactly three probes before being inserted into the table is exactly $\frac{2}{m(m-1)}$.
- C. We use a hash table of size m with open addressing to hash n items. Under the uniform hashing assumption, the expected cost to insert another element into the table is at most $1+\alpha$, where $\alpha=n/m$ is the average load.
- D. Linear probing is equivalent to double hashing with a secondary hash function of $h_2(k) = 1$.



Consider a hash table of 9 slots implemented with linear probing. Suppose we insert 2 elements in a sequence to a hash table with a simple uniform hashing assumption. What is the probability that we end up with 2 consecutive slots of the hash table filled?

Slot i and (i + 1) mod m are defined to be consecutive slots.

- A. 1/2
- B. 1/3
- C. 1/4
- D. 2/3

D.

https://gateoverflow.in/quiz/results.php

Your Answer: B **Correct Answer: B** Correct Discuss

Q #11 **Multiple Choice Type** Algorithms Award: 2 Penalty: 0.67

An open-addressing hash table(with m slots) that resolves collisions using linear probing is initially empty. Key k1 is inserted into the table first, followed by k2, and then k3 (the keys themselves are drawn randomly from a universal set of keys).

What is the probability that searching for k1 takes exactly two probes?

- A. $(m-2)/m)^3$
- B. $3/m^2$
- C. $2/m^2$
- D. None of these

Your Answer: D **Correct Answer: D** Correct Discuss

Q #12 **Multiple Choice Type** Award: 2 Penalty: 0.67 Algorithms

Suppose you insert three keys into a hash table with m slots. Assuming the simple uniform hashing assumption, and given that collisions are resolved by chaining, what is the probability that both slots 0 and 1are empty?

- A. $((m-2)/m)^3$
- B. $((m-3)/m)^3$
- C. $(m/m-1)^3$
- D. $((m-2)/(m-1))^3$

Your Answer: A **Correct Answer: A** Correct Discuss

Multiple Choice Type Award: 2 Penalty: 0.67 Algorithms

Consider a hash table with m slots that uses chaining for collision resolution. The table is initially empty. What is the probability that after 4 keys are inserted that at least a chain of size 3 is created?

- A. $\frac{4m-3}{m^3}$ B. m^{-4}
- C. $m^{-3}(m-1)$
- D. m^{-2}

Your Answer: A **Correct Answer: A** Correct **Discuss**

Q #14 Award: 2 Algorithms **Numerical Type** Penalty: 0

Suppose we implement linear probing with the hash function $h(x) = x \mod 9$. After inserting 7 keys, the table is shown as below:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|----|---|----|---|----|---|----|---|
| 9 | 18 | | 12 | 3 | 14 | 4 | 21 | |

How many total insertion orders of keys are possible to reach the above state of the hash table?

For example, two possible insertion orders are -

- \bullet 12, 14, 3, 9, 4, 18, 21 and
- 9, 12, 14, 3, 4, 21, 18





Consider a hash table with n buckets, where external (overflow) Chaining is used to resolve collisions. The hash function is such that the probability that a key value is hashed to a particular bucket is 1/n.

The hash table is initially empty and k distinct values are inserted in the table. What is the probability that the first collision occurs at the $k^{\rm th}$ insertions?

A.
$$\dfrac{(n-1)!}{(n-k)! \; n^{k-1}}$$
B. $\dfrac{(n-1)!}{(n-(k-1))! \; n^k}$
C. $\dfrac{n!(k-1)}{(n-(k-1))! \; n^k}$
D. None of these



You're doing good, you can target above 70 percentage!

Copyright & Stuff

5/5