HackerRank For Work

Spring_2018_INFO6205_Se... 35 minutes

Question - 1

SCORE: **20 points**

Linear Probing

Suppose we are using Hash(k) = 3 * k % 13, and an array of size 13 as an Hash Table, what's the result after put the below number into the hash table if we use linear probing? (* represent there is no value in the hash table) Number in order: 22 -> 40 -> 36 -> 55 -> 24 -> 27 -> 28

- * 22 * 40 36 27 * 24 * 55 28 * *
- 22 * 40 36 27 28 24 * 55 * * * *
- 22 * 27 36 28 * 24 * 55 * * * *
- * 22 * 40 36 27 28 24 * 55 * * *
- * 22 * 40 27 36 * 24 * 55 * * *
- * 22 * 27 36 28 * 24 * * * * *
- * 22 * 40 36 * * 24 * 55 * * *

Question - 2

SCORE: **20 points**

Hash Function1

Suppose we have an instance of a class which contains 2 attributes: Name and ID. We manually override the hash function and our own hash function will calculate both attributes (eg. name.hashCode() + ID). First, we add this instance into an empty HashSet. Then we modify the ID of this instance (eg. setID(xxx)). This time, when we call contains function (contains(instance)), what will we get?

- Null
- True
- False
- Runtime Error

Question - 3

SCORE: **20 points**

Hash Function2

Based on the above question, what if we use the default hash function (not override with our own hash function), what will we get?

- Null
- True

- False
- Runtime Error

Question - 4

SCORE: **20 points**

Bonus!

Briefly explain why for Q2 and Q3

Question - 5

SCORE: **40 points**

Coding

Given a string, find the first non-repeating character in it and return it's index. If it doesn't exist, return -1. The string may contain upper case, lower case, number, and symbol.

Example:

Str = “qwerty” return: 0

Str = “qqwer” return: 2

Str = “qqq” return -1