LE/EECS 4101 GS/EECS 5101
Advanced Data Structures

Quiz 2

Shahin Kamali

York University

March 22nd, 2023

Write your name and student id here:

"The deep roots never doubt spring will come ..." *Marty Rubin*

- **Do not open this booklet until instructed**.
- You are NOT allowed to use any printed/written material.
- Please **turn off your cell phones** and put them in your bags.
- Manage your time. We start the exam at 5:30 and end at 6:00. **You have 30 minutes**. **Don't waste too much time on a single question.**
- The exam is printed **double-sided**. There are **6 pages**, including this cover page and a blank page (use it if you need more space). You must submit ALL pages.

1. True/False and Multiple-Choice Questions (15 marks)

Provide your short answers in the provided boxes. There is no need to justify your answers.

(a) True or False: For performance improvement, the sizes of hash tables are best to be powers of 2. [False]

Answer: As discussed in the class, the hash table size is best to be a prime that is not close to a power of 2.

(b) True or False: The load factor can be more than 1 when handling collisions with chaining. [True]

Answer: Unlike open-addressing, the load factor in chaining can be larger than 1.

(c) Tue or False: after rehashing, it is necessary to modify the hash function. [True] **Answer:** Since the hash table size is changed, the hash function must change.

(d) Assume a hash scheme in which keys are selected uniformly at random from the Universe set $U = 1, 2, 3, \ldots, 600$. Consider the following hash functions. Which hash function is better? (choose one answer)

a) $h_1(k) = 2k \bmod 8$ ☐

b) $h_2(k) = 3k \bmod 8$ ☐

c) $h_3(k) = 4k \bmod 8$ ☐

d) $h_4(k) = 6k \bmod 8$ ☐

e) all hash functions are equally good.

Answer: The answer is (b).
If we use $h_1(k)$, we will only use indices $0, 2, 4, 6$. If we use $h_2(k)$, we will use all indices $0, 1, 2, 4, 5, 6$. If we use $h_3(k)$, we will only use indices $0, 4$. If we use $h_4(k)$, we will only use indices $0, 2, 4, 6$.

(e) True or False: In a Quad tree of n points, where n is a large number, it is possible that one of the children of the root be empty (null). [True]

(f) True or False: The height of the kd-tree for any set of n points is $O(\log n)$. [True].

Answer: Kd-trees are balanced (since at each level, you find the medial and partition the points into two sets of equal size.)

(g) True or False: Range query time in range trees is asymptotically faster than in kd-trees. [True].

Answer: In 2d-range trees, the range queries take $O(k + \log^2 n)$, which is faster than $O(k + \sqrt{n})$ of kd-trees.

2. Hashing (3 + 3 + 4 marks)

Write down the answer to the following questions. Unless otherwise mentioned, no justification is needed.

(a) We apply the Horner's rule with $m = 10$ to find the hash value for "PdP" in a hash table of size 100. Recall the following code for applying Horner's rule. The ASCII value of P and d are respectively 80 and 100. Write down the returned index.

```
static int hashFn(String key) {
      int hashCode = 0;
      int m = 10;
      for (i = 0 ; i < key.length() ; i++)
         hashCode = m * hashCode + (int) key.charAt(i);
      return hashCode mod 100;
}
```

80

Answer: Initially, hashCode is 0. After the first iteration, it will become 80; then it becomes $80 * 10 + 100 = 900$. Finally, it becomes $900 * 10 + 80 = 9080$. The ouput will be 9080 mod 100 = 80.

(b) Given a "large" array L of n distinct integers and a "small" array S of $\lfloor \log n \rfloor$ distinct integers, we want to report all numbers that appear in both L and S. For example, for $L = [3, 10, 5, 8, 14, 2, 1, 20]$ and $S = [11, 20, 3]$, the output must be $\{3, 20\}$ (reported in any order).
The following two solutions both use **Cuckoo hashing** to handle collisions in hash table H.

- **Solution 1:** scan L and insert all its items in the hash table H; then scan S and search for any item in H.
- **Solution 2:** scan S and insert all its items in the hash table H; then scan L and search for any item in H.

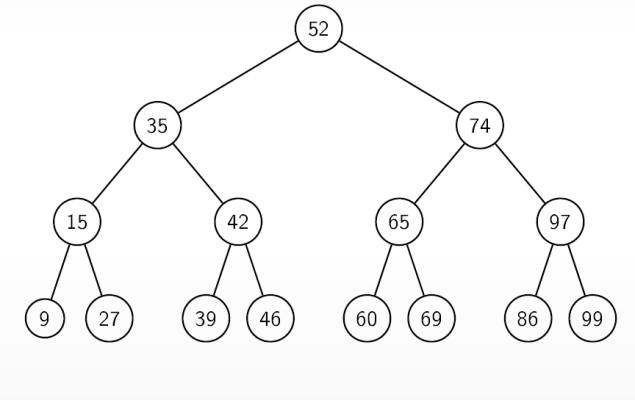
Specify which solution is better. Justify your answer in one or two sentences.

Answer: Solution 2 is better. In Cuckoo hashing, search time is $O(1)$ (even if the uniform hashing assumption does not hold), while insertion could be slower. Thus, a solution with a smaller number of insertions is preferred.

(c) Consider a hash table dictionary with universe $U = \{0, 1, 2, \ldots, 24\}$ and size $M = 5$. If items with keys $k = 21, 3, 16, 1$ are inserted in that order, draw the resulting hash table if we resolve collisions using **Cuckoo hashing** with $h_1(k) = k \bmod 5$ and $h_2(k) = \lfloor k/5 \rfloor$.

3. Multidimensional Dictionaries (3 + 4 + 3 marks)

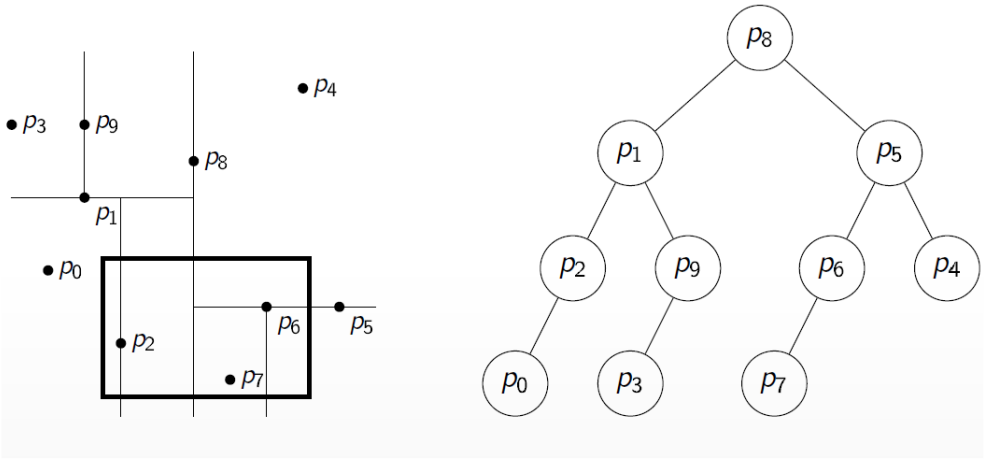
(a) Specify all **outside** nodes when we search for items in the range $(40, 80)$ in the following 1-dimensional binary search tree.



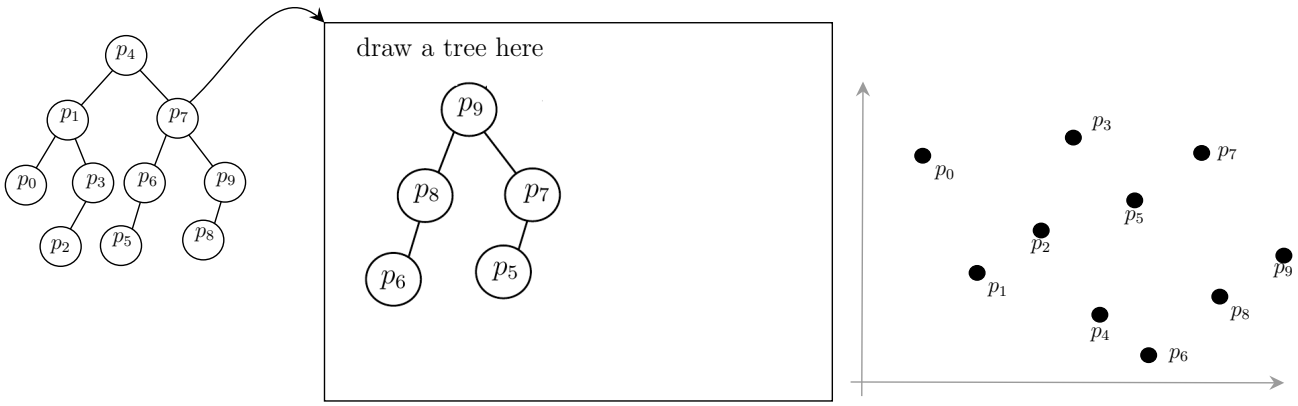
9, 15, 27, 99

Answer: Outside nodes are the ones which are dismissed (not reported) without being examined. In this case, boundary nodes are $52, 35, 42, 39, 74, 97, 86$, and inside nodes are $46, 60, 65, 69$

(b) Specify the nodes of the following kd-tree that are **examined** when searching for the points inside the highlighted query box. $p_8, p_1, p_2, p_0, p_5, p_6, p_7, p_4$



(c) Draw the secondary tree pointed by the node for p_7 in the range tree associated with the following points.



Use this blank page for your draft work.