IMPORTANT:
- Upload this PDF with your answers to Gradescope by 11:59pm on Monday Sept 30, 2019.
- Plagiarism: Homework may be discussed with other students, but all homework is to be completed individually.
- You have to use this PDF for all of your answers.

For your information:
- Graded out of 100 points; 4 questions total
- Rough time estimate: ≈1-4 hours (0.5-1 hours for each question)

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cuckoo Hashing</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>B+Tree</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>Extendible Hashing</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Suffix Trees</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

Number of Days this Assignment is Late:

Number of Late Day You Have Left:
Question 1: Cuckoo Hashing ........................................... [20 points]
Graded by:
Consider the following cuckoo hashing schema:

1. Both tables have a size of 4.
2. The hashing function of the first table returns the lowest two bits: \( h_1(x) = x \& \ 0b11 \).
3. The hashing function of the second table returns the next two bits: \( h_2(x) = (x \gg 2) \& \ 0b11 \).
4. When replacement is necessary, first select an element in the second table.
5. The original content is shown in Figure 1.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Table 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Initial contents of the hash tables.

Use the following template to answer the questions: https://cmudb.io/fall2019-hw1.
(a) [4 points] Insert keys 12 and 10. Draw the resulting two tables.
(b) [4 points] Then delete 14, and insert 8. Draw the resulting two tables.
(c) [6 points] Finally, insert 28. Draw the resulting two tables.
(d) [6 points] What is the smallest key that potentially causes an infinite loop given the tables in (c)

[ ] 0 [ ] 2 [ ] 5 [ ] 6 [ ] 7 [ ] 9 [ ] None of the above

**Solution:**

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Table 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>12</td>
</tr>
</tbody>
</table>

Question 1 continues...
### b) Table 1

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2

<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
</tr>
<tr>
<td>12</td>
</tr>
</tbody>
</table>

### c) Table 1

<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
</tr>
<tr>
<td>10</td>
</tr>
</tbody>
</table>

### Table 2

<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>28</td>
</tr>
</tbody>
</table>
Question 2: B+Tree ............................................................ [45 points]

Graded by:

Consider the following B+ tree.

![B+ Tree Image]

Figure 2: B+ Tree of order \( d = 4 \) and height \( h = 2 \).

When answering the following questions, be sure to follow the procedures described in class and in your textbook. You can make the following assumptions:

- A left pointer in an internal node guides towards keys \(<\) than its corresponding key, while a right pointer guides towards keys \(\geq\).
- A leaf node underflows when the number of keys goes below \(\lceil \frac{d-1}{2} \rceil\).
- An internal node underflows when the number of pointers goes below \(\lceil \frac{d}{2} \rceil\).

Use the following draw.io template for your answers:
https://cmudb.io/fall2019-hw2

(a) [15 points] Insert 10* into the B+tree. Draw the resulting tree.

(b) [5 points] How many pointers (parent-to-child and sibling-to-sibling) do you chase to find all keys between 5 and 15?

- \( \square \) 2
- \( \square \) 3
- \( \blacksquare \) 4
- \( \square \) 5
- \( \square \) 6
- \( \square \) 7

(c) [15 points] Then delete 23*. Draw the resulting tree.

(d) [10 points] Finally insert 4* and delete 11*. Draw the resulting tree.

Solution: a)

![Solution Diagram]

b)
c)
Question 3: Extendible Hashing ................................. [25 points]
Graded by:

Consider an extendible hashing structure such that:

- Each bucket can hold up to two records.
- The hashing function uses the lowest $g$ bits, where $g$ is the global depth.

(a) Starting from an empty table, insert keys 15, 3, 7, 14.
   i. [3 points] What is the global depth of the resulting table?
      □ 0 □ 1 □ 2 ■ 3 □ 4 □ None of the above
   ii. [3 points] What is the local depth the bucket containing 14?
       □ 0 ■ 1 □ 2 □ 3 □ 4 □ None of the above
   iii. [3 points] What is the local depth of the bucket containing 3?
        □ 0 □ 1 □ 2 ■ 3 □ 4 □ None of the above

(b) Starting from the result in (a), you insert keys 1, 9, 23, 11, 17.
   i. [4 points] Which key will first cause a split (without doubling the size of the table)?
      □ 1 □ 9 □ 23 □ 11 ■ 17 □ None of the above
   ii. [4 points] Which key will first make the table double in size?
       □ 1 □ 9 ■ 23 □ 11 □ 17 □ None of the above

(c) Now consider the table below, along with the following deletion rules:
   1. If two buckets have the same local depth $d$, and share the first $d - 1$ bits of their indexes (e.g. 010 and 110 share the first 2 bits), then they can be merged if the total capacity fits in a single bucket. The resulting local depth is $d - 1$.
   2. If the global depth $g$ becomes strictly greater than all local depths, then the table can be halved in size. The resulting global depth is $g - 1$.

![Figure 3: Extendible Hash Table along with the indexes of each bucket](image)

Figure 3: Extendible Hash Table along with the indexes of each bucket

Question 3 continues...
Starting from the table above, delete keys 2, 7, 13, 15, 29.

i. [4 points] Which deletion first causes a reduction in a local depth.
   □ 2 □ 7 ■ 13 □ 15 □ 29 □ None of the above

ii. [4 points] Which deletion first causes a reduction in global depth.
    □ 2 □ 7 ■ 13 □ 15 □ 29 □ None of the above
Question 4: Suffix Trees ........................................... [10 points]
Graded by:
Consider the following suffix tree for unsigned 32-bit integers.

(a) [3 points] Which of the following elements belong to the suffix tree. Select all that apply.
- 0x45BD0000
- 0x0000CAAC
- 0xFFAAAA00
- 0xACCA0000
- 0xBD000000
- None of the above

(b) [7 points] Insert the key 0x00FFAABB. Draw the resulting tree using this template: https://cmudb.io/fall2019-hw4.

Solution: b)