Readings

- Lecture Notes Chapter 23: Hashing

Problems

Problem 1
With \( n \) distinct balls in \( m \) distinct bins what is the probability that no bucket has more than 1 ball? You may assume that \( n \leq m \).

Problem 2
Assume we have a hash table \( T \) of size 10 that uses linear probing and has hash function \( h(x) = x \mod 10 \). We insert 6 numbers into \( T \) and we get the below table:

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>42</td>
<td>23</td>
<td>34</td>
<td>52</td>
<td>46</td>
<td>33</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What is one possible order that we could have inserted these elements to get this result? How many probes would be required for inserting 13 in the table?

Problem 3
Design an algorithm that determines if two lowercase words are anagrams of each other in expected \( O(n) \) time. Note: A string \( A \) is an anagram of another string \( B \) if \( A \) is a permutation of \( B \). Can you do it in worst case \( O(n) \) time?

Problem 4
You are given an array \( A \) containing distinct randomly assorted integers. Your goal is to find two elements in the array whose sum is \( k \) in \( O(n) \) expected time.

Problem 5
How would you detect a cycle in a linked list of distinct elements in expected \( O(n) \) time? Can you do it in constant space?