

MTH 4320 Homework 6

Yaohui Wu

March 20, 2024

Problem 1

Solution. If the cell is empty then we can insert with constant time. If the cell is not empty then there is a collision so we need to iterate over the other cells to find an empty one. Therefore, the time complexity of insertion without collision is $O(1)$ and with collision is $O(n)$. If the element is in cell i then we can search in constant time. If it is not in cell i then we need to iterate over the other cells to find it. Therefore, the time complexity of searching without collision is $O(1)$ and with collision is $O(n)$. ■

Problem 2

Solution. The algorithm is:

1. Let H be a hash table with k cells. The time complexity is $O(k)$.
2. For every element in L : Insert the element to the k th cell where k is the key of the element. The time complexity is $O(1)$. If the cell is not empty then chain the element. The time complexity is $O(n)$.
3. Make a new sorted L by appending the elements in every cell of H in order. We have n elements and k keys so there are at most $\frac{n}{k}$ values in every cell of H . The time complexity is $O(k) \cdot O(\frac{n}{k}) = O(k \cdot \frac{n}{k}) = O(k)$.

The time complexity of the algorithm is $O(n + k)$. ■

Problem 3

Solution. The algorithm using the sliding window approach is:

- 1.

The time complexity of the algorithm is $O(n)$. ■

Problem 4

Solution. The algorithm using the sliding window approach is:

- 1.

The time complexity of the algorithm is $O(n)$. ■