Sorting

Given an array $A$ of $n$ bit strings, all of length $m$, describe an algorithm to sort them in lexicographical order in $O(mn)$ time.

Quadratic Probing

Define $h(k, i) = h(k) + \frac{i(i+1)}{2} \mod M$ and let $M = 2^p$ for some non-negative integer $p$.

1. Write the probe sequence for $h(k) = 0$ and $p = 3$ starting from $i = 0$ to $i = M - 1$

2. Prove that all entries in the probe sequence are different – therefore the probe sequence will hit an empty slot

Ordered Hashing

For this question we will look at a modified version of open addressing such that, while searching in the hash table, the sequence of keys encountered during the search has some ordering property (though the sequence is not necessarily in sorted order). To achieve this property, we use a modified insert routine: when inserting key $k$, follow its probe sequence, and if you ever encounter a key $k'$ with $k' > k$, then swap to put $k$ into this position, and proceed to insert $k'$.

1. Carry out this idea for linear probing using $M = 10$, the hash function $h(k, i) = (k + i) \mod 10$, and the insertion sequence, 31, 26, 16, 23, 11, 30, 20.

2. Argue that for searching for any key, the keys encountered before it in its probe sequence are all smaller than it. You may assume that no items will ever be deleted from the hash table.

Amortized Analysis

Do Problem 2 from the midterm.