University of Waterloo
CS240 Spring 2020
Assignment 4 Post Mortem

We normally publish a post-mortem for an assignment after it has been marked and released. Here is a list of common errors provided by the graders for assignment 4. Please look at this feedback, as well as the individual feedback provided for your assignment, to get an idea of where you can improve in the future.

Problem 1

Many students gave the wrong height of O(n). The height of T can't necessarily be O(n) since the values in T must be $\leq n^f$ and are represented in base 2^t. Suppose we take the largest value n^f, then the length of its 2^t representation is $\log_{2^t} n^f$ which can be simplified to $\frac{f}{t}log(n)$. This is an upper bound on the length of one value, and the height of T is determined by the value with the largest length. Therefore the height of T should be in O($\frac{f}{t}log(n)$).

Most student got the correct answer of O(n2^t) for space. Some students stated that the number of nodes in T is n since there are n keys, which is incorrect. The n keys corresponds to the n leaves in T, but we also have the n-1 internal nodes. Some students gave an answer of O((2n-1)2^t). Note that for big-O notations we only keep the dominant terms and omit the coefficients.

Most students were able to state that the search time is proportional to the height.

Problem 2

This question was done well. Some students missed the labels in their tries and some gave incorrect compressed tries.

For part (f), some students did not use the Induction Hypothesis in their Inductive Steps. Several students who did invoke their Induction Hypothesis in their Inductive Step were not clear about why the height of the trie will actually increase when adding the elements $2^i, 2^i + 1, ..., 2^{i+1} - 1$.

Problem 3

For part(a), some students stored the values in the table instead of using linked lists. In chaining we store the value in linked list and insert the nodes after the correct indexes. The

hash table itself only stores pointers pointing to the start of each linked list.

For part (d), a few students did not use the second hash table for cuckoo hashing.

Problem 4

This question was done well.

For 4(b) some students did not explain how to calculate c_n using the indicator variables.

Problem 5

5(a) was done well.

For 5(b), most students got the correct idea. Some students did not mention that ultimately we will call insert on a key that is greater than any key in the table and in this case we will perform a regular probing.

For 5(c), most students gave a correct hash function and a correct hash table. Some students misunderstood the question; we want to show that inserting one key in the hash table would take $\Omega(n^2)$ time, not filling the hash table with a sequence of keys.