CSCE 313-511 MP1 Report
Colin Banigan and Jessica Fang

Part 1:

1) **Do you notice any wastage of memory when items are deleted? If so, can we avoid such wastage and how?**

Initially every time a node was deleted, the key value was set to a negative number which cannot be seen by the lookup function. Due to this, the node, although it cannot be seen by the program, was still located in memory. With enough inserts and deletes, the memory allocated could have been depleted while still having an empty list.

Instead of having this situation, we could create a case in the delete function which gives us a way to see if a node has been deleted. When insert happens, instead of "directly" appending to the end, the function first checks to see if there is a deleted node. If there is a deleted node, the node at the end of the linked list is re-mapped to point back to the deleted node. This deleted node is then replaced with the insert information, and changes the node prior to the inserted node to point to the node after the deleted node.

2) **Can you think of a scenario, where there is space in the memory, but no insertion is possible?**

If value_len is greater than the number of bytes per node minus the byte size of the arguments. Since the byte size of the arguments is 4, 8, and 4 respectively, we would check:

$$value\_len > (b - 16)$$

3) **What is the maximum size of the value when the pointers are 8 bytes?**

Continuation from the last question which we answered indirectly, the maximum value would be:

$$b - 16$$

Part 2:

1) **Derive a general expression for the range of numbers that go into the i-th tier.**

Assuming the value "i" starts at 0 and goes until (numTiers - 1):

$$[i * (2^{31} - 1)/numTiers , (i + 1) * (2^{31} - 1)/numTiers]$$