Homework 4
Due: Wednesday, July 2nd by 11:59 pm Online.

1. [60 Pts] Hash Table Concepts and Implementation

1.1 (*5 points*) Assume that a hash table has 9 slots and the hash function is $h(k) = k \bmod 9$. Demonstrate the insertion into the hash table of the following sequence of keys with collision resolved by chaining: 5, 28, 19, 15, 20, 33, 12, 17, 10, 13

1.2 (*5 points*) Consider a hash table of size m=1000 and a corresponding hash function $h(k)$ given by the multiplication method for $A \approx (\sqrt{5} - 1)/2 = 0.6180\ldots$ (as discussed in the lecture slides). Compute the locations to which the following sequence of keys are mapped: 61, 62, 63, 64, 65.

1.3 (50 points) Suppose you are implementing a dynamic set of student records as a hash table. Each record has an integer key. Key can have values from 0 through 65,536 and no two records can have the same key values. In addition to the key, each record has following information.

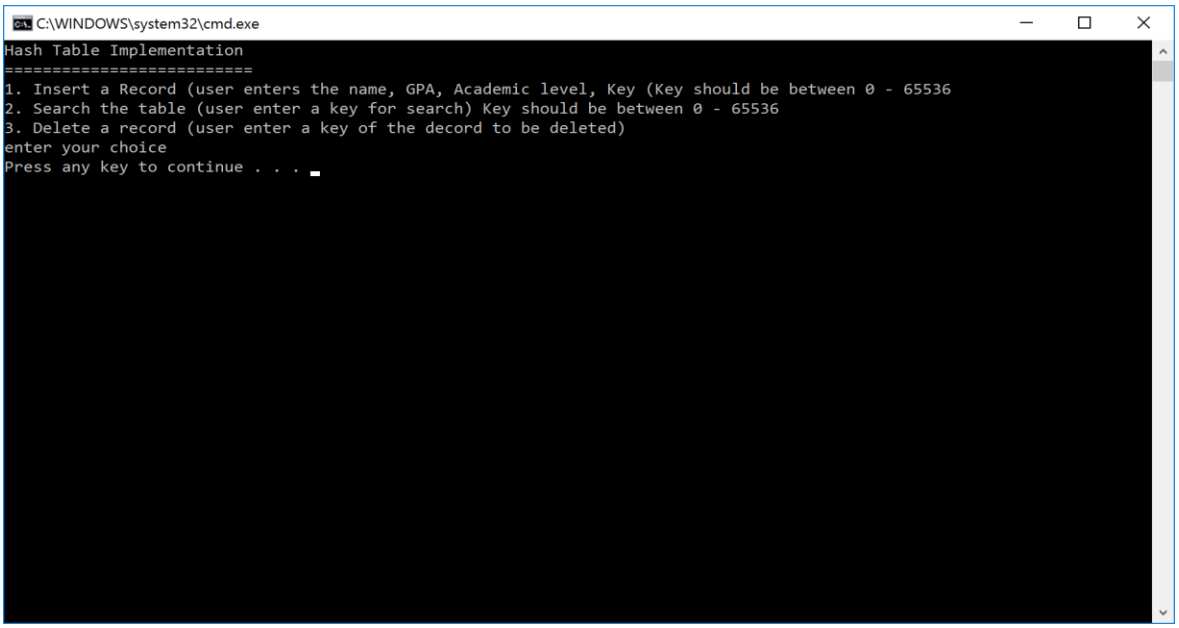
Name:
GPA:
Academic level:

Even though the key can take a value between 0 and 65536, this university can have max 10000 students at a given time.

Hash Table Implementation Details:
Assume that the hash key is **k mod m and using chaining** in case of a collision. Also, the size (m) of the hash table is 1000. Also, you can assume that keys for student are generated using a random uniform distribution function.

Write a C or C++ program that Implements the hash table construction for the above scenario and then implement the following three functions

a) INSERT(T, x) // insert the student record x to the table
b) DELETE(T, x) // delete the student record x from the table
c) SEARCH(T, k) //search key k in the hash table



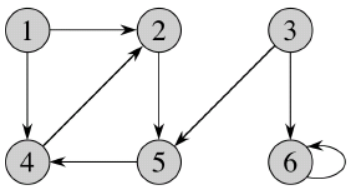
2. [15 Pts] Red-Black tree is a binary search tree with following properties.

a. Every node is either red or black.
b. Every leaf node (nil) is black.
c. If a node is red, then both of its children are black.
d. All paths from a node to its descendant leaves contains the same number of black nodes.
e. The root node is black.

a) As we have learned, for Red-Black trees, the ***black-height*** of a node x, bh(x), is the number of black nodes (including the leaf node) on the path from x to any leaf, not counting x. Prove that the subtree rooted at any node x contains $\geq 2^{bh(x)}$ -1 internal nodes. [Hint: Do induction on the height of x.]

b) Prove that the insert operation of the red-black tree is always log(n)

3. [25 Pts] Graph Representation: Consider the following graph G and answer questions given below.



a) Represent the above graph using adj-matrix and adj-list techniques.
b) Write a pseudo-code that implements adj-matrix and adj-list techniques above (create the adj-list and adj-matrix representation)

SUBMISSION INSTRUCTIONS

A) Complete code (C/C++) that implement 1.3
B) Answers to questions 1.1, 1.2, 2 and 3 as separate PDF documents

Make a zip file called hw4.zip and submit online