1. Consider the insertion of items with the following keys (in the given order) into an initially empty RedBlack tree: 30, 40, 24, 58, 26, 11, 13, 58, 62, 70, 94, 2. Draw the status of the tree after each insertion (i.e. draw the sequence of trees).

2. A certain Professor Amongus claims that the order in which a fixed set of elements is inserted into a RedBlack tree does not matter (i.e. the same tree results every time). Give a small example that proves Professor Amongus wrong.

3. Is there a heap $T$ storing seven distinct elements such that a preorder traversal of $T$ yields the elements of $T$ in sorted order? How about an inorder traversal? How about a postorder traversal? Give an example of each or state that none exist for any given traversal.

4. Alice says that a hash table with collisions handled using separate chaining can have a load factor greater than 1. Bob says that this is impossible. Who is right and why?

5. Draw the 11-item hash table resulting from inserting 12, 44, 13, 88, 23, 94, 11, 39, 20, 16, 5 using the hash function $h(i) = (2i + 5) \mod 11$ and assuming that collisions are handled by separate chaining.

6. Draw the resulting hash table of Exercise 5, assuming that collisions are handled using linear probing.

7. Suppose you would like to build a hash table for images, where the key for each image is a “thumbnail” image of 75x75 pixels, with each pixel being one of 256 possible colors. Describe a hash function for a collection of such images. Your hash function should be fast to compute and it should strive to map different images to different hash values. In particular, reflections and $90^\circ$ rotations of the same image should, in general, map to different hash values.