1. Consider an unsorted collection of data of size $n$ where the collection can be implemented either as an array or a linked list.
   (a) How long does it take to search the data for an item, insert an item and delete an item assuming an array implementation?
   (b) Repeat (a) assuming a linked list implementation.

2. Repeat 1. assuming a sorted collection of data and both insertion and deletion operations yield a new sorted collection. Comment and compare the different search, insert and deletion algorithms.

3. Implement an insert sort method on your collection to ensure that the collection remains sorted. Assume a linked-list implementation. Discuss shortcomings in your design (to be debugged at a later date) and use a diagram to illustrate what you have done (or to show what you would like to do except you can’t figure out the code!).

4. The storage requirements for a typical RISC processor are: 4 bytes for integers, floats and pointers and 8 bytes for doubles. If 1140 bytes are required to store 1000 bytes of data we define the overhead as being 14%.
   (a) What is the overhead for storing 100 items and 1000 items of integers using the collection structure implementation of an array as discussed in class?
   (b) Repeat (a) for the collection structure implementation of a linked list.
   (c) Repeat (a) and (b) for storing items of doubles
   (d) Repeat (a) and (b) for storing items of:

```c
struct {
    int x, y;
    double z[20];
}
```