```
/*
 *  Author: Scot Matson {009602502}
 *  Course: CS49C - Section 01
 *  Assignment: 09
 *  Date: 11/22/14
 *
 */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

struct ListNode {
  unsigned int data;
  char name[16];
  char grade;
  struct ListNode *next;
};

void insertList(struct ListNode *lptr, unsigned int
 idata, char *iname, char igrade) {

  struct ListNode *curr_node = lptr->next;
  struct ListNode *new_node  = (struct ListNode*) m
alloc(sizeof(struct ListNode));

  new_node->data = idata;
  strcpy(new_node->name, iname);
  new_node->grade = igrade;

  if (lptr->next == NULL) {
    new_node->next = lptr->next;
    lptr->next = new_node;
  }

  else if(new_node->data < curr_node->data) {
    new_node->next = curr_node;
    lptr->next = new_node;
  }
  else {
    while (new_node->data > curr_node->next->data)
{
      curr_node = curr_node->next;
```

```
    }
    new_node->next = curr_node->next;
    curr_node->next = new_node;
  }
}

void printList(FILE *fpo, struct ListNode *lptr) {
  lptr = lptr->next;
  while (lptr->next != NULL) {
    fprintf(fpo, "%09u \t%s \t%c\n", lptr->data, lp
tr->name, lptr->grade);
    lptr = lptr->next;
  }
}

void freeList (struct ListNode *lptr) {
  // will free memory locations here
  struct ListNode *temp_node;
  lptr = lptr->next;

  while (lptr->next != NULL) {
    temp_node = lptr;
    lptr = lptr->next;
    free(temp_node);
  }
}
```