IC210 Lab47 Name: _____

Read this assignment in its entirety before starting. Create a MSVC++ project and add the four files provided on the Class 47 web page: **list.h**, **list.cpp**, **main.cpp**, and **data.txt**. The only file you should modify is list.cpp, which you are to complete by implementing the following functions (the first four we should have completed in class):

```
// Adds a node containing n to the front of linked list L
void addFront( int n, node_t *L );

// Prints linked list L
void print( node_t *L, ostream& out );

// Deletes the first node in linked list L
void deleteFront( node_t* &L );

// Deletes linked list L
void deleteList( node_t* &L );
```

```
// Uses recursion to print linked list L
void recprint( node_t *L, ostream& out );

// Uses recursion to print linked list L in reverse order
void recprintrev( node_t *L, ostream& out );

// Returns the length of linked list L
int length( node_t *L );

// Adds a new node containing data value n to linked list L
// The new node will be the last node in the list
void addBack( int n, node_t* &L );

// Returns a pointer to the node in linked list L that has
// the minimum data value of all nodes in the list.
// If the list is empty, returns a null pointer
node_t *min( node_t *L );

// Swaps the values in data members pointed to by p and q
void swap( node_t *p, node_t *q );

// Performs a selection sort on the data values in linked list L
void selectionSort( node_t* &L );

// Inserts a new node with data value n into the linked L.
// L is in sorted both before and after insertion
void insertInOrder( int n, node_t * &L );

// Performs an insertion sort on the data values in linked list L
void insertionSort( node_t* &L );
```

Note that some of these functions already have a function body, but the code is wrong.

Turn in: (a) This sheet with your name filled in, to which is stapled
(b) A hardcopy listing of list.cpp
(c) A screen snapshot of the output of the main program.