Background:
In previous assignments, you have stored data in arrays. An array is useful when the size of your input set is known at the beginning of your application. After pointers were introduced, it became possible to declare arrays at runtime—i.e., prompt the user for the size of an array.

To add or remove elements from an array, the array must be resized. After the array has been resized, the elements must be rearranged. In an array, this is an expensive operation. To remove one element, all subsequent elements must be shifted.

A linked list is a collection of nodes wherein each node points to the next node in the list. The last node in a linked list points to NULL. Linked lists allow nodes to be added and removed. When manipulating a linked list, all operations involve updating pointers.

Description:
In this assignment, you will create a linked list. You have been provided with three structs. The LinkedList contains a head pointer, tail pointer, and num_nodes counter. The pointers—i.e., head and tail—reference the beginning and end of the linked list respectively. The num_nodes is simply a counter for the number of nodes in the list.

The Node struct contains a pointer to data. Data refers to the information—in this assignment, Treasure—contained within the node. The next pointer refers to the subsequent node in the linked list.

Instructions:
1. Preliminary Instructions
   - Read this prompt completely.
   - Read the provided template—including all comments.
2. Structs
   - Familiarize yourself with the provided structs.
   - The LinkedList and Node structs are described in the Description section.
   - The Treasure struct contains:
     - Location—where a treasure was found.
     - Name—the treasure that was found.
3. **Function: main**
   - Complete the input loop—listed as **Modification 1**.
     - This loop reads each line of the input file, creates a new treasure, and invokes the `append` function.
     - Complete this loop:
       - Store the location in `to_add->location`.
       - Store the name in `to_add->name`.
   - Uncomment the `removeNode` function invocation.
     - Wait until after you have completed the `appendNode` function.
   - Complete the second loop—listed as **Modification 2**.
     - Store the name in `to_add`.
     - Store the location in `to_add`.
     - Invoke the `appendNode` function.
     - Hint: Use the arrow operator to access attributes of a Treasure.

4. **Function: appendNode**
   - The function body has been provided.
   - The code to add the first node to the linked list has been provided.
   - Complete the else block.
     - Add the new node to the linked list.
     - Update the tail pointer.
     - This should be between 2-4 lines.

5. **Function: removeNode**
   - This function is complete.
   - Familiarize yourself with the logic in this function.
   - Make no changes to this function.

6. **Function: deleteLinkedList**
   - This function is complete.
   - Familiarize yourself with the logic in this function.
   - Make no changes to this function.

7. **Function: printTreasureList**
   - This function should print the entire linked list as a table—including a heading.
   - A list_iterator and index have been provided.
   - The table heading has been provided.
   - This list_iterator has been initialized.
   - Complete the while loop.
     - Each loop iteration must print exactly one treasure.
     - Your output must match the sample output.
Sample Output:

---

Enter the input filename: A9-input.txt

Original Treasure List

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Web</td>
<td>PSP</td>
</tr>
<tr>
<td>2</td>
<td>Ta-Office</td>
<td>N64</td>
</tr>
<tr>
<td>3</td>
<td>Dragas</td>
<td>NES</td>
</tr>
<tr>
<td>4</td>
<td>Dragas</td>
<td>SNES</td>
</tr>
<tr>
<td>5</td>
<td>Dragas</td>
<td>Wii</td>
</tr>
<tr>
<td>6</td>
<td>Spong</td>
<td>Nothing</td>
</tr>
</tbody>
</table>

Treasures Found: 6
Remove which Node: 6

Pruned Treasure List

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Web</td>
<td>PSP</td>
</tr>
<tr>
<td>2</td>
<td>Ta-Office</td>
<td>N64</td>
</tr>
<tr>
<td>3</td>
<td>Dragas</td>
<td>NES</td>
</tr>
<tr>
<td>4</td>
<td>Dragas</td>
<td>SNES</td>
</tr>
<tr>
<td>5</td>
<td>Dragas</td>
<td>Wii</td>
</tr>
</tbody>
</table>

Treasures Found: 5

New Treasure

**What did you find?** : Carpet
**Where was it found?** : Spong

**What did you find?** : 3DSXL
**Where was it found?** : Dragas

Final Treasure List

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Web</td>
<td>PSP</td>
</tr>
<tr>
<td>2</td>
<td>Ta-Office</td>
<td>N64</td>
</tr>
<tr>
<td>3</td>
<td>Dragas</td>
<td>NES</td>
</tr>
<tr>
<td>4</td>
<td>Dragas</td>
<td>SNES</td>
</tr>
<tr>
<td>5</td>
<td>Dragas</td>
<td>Wii</td>
</tr>
<tr>
<td>6</td>
<td>Spong</td>
<td>Carpet</td>
</tr>
<tr>
<td>7</td>
<td>Dragas</td>
<td>3DSXL</td>
</tr>
</tbody>
</table>

Treasures Found: 7