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.

```
1 ─ 2 ─ 3 ─ 4 ─ NULL
```

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
   - Invoke parseFile.
   - Invoke AddTreasures.
   - Use getline to retrieve a location.
   - Invoke examineLocation.
4. **Function:** addTreasures
   - Complete the first user prompt—read item name
     - Use getline to read the entire line—including spaces.
   - Complete the second user prompt—read item location
     - Use getline to read the entire line—including spaces.
   - Invoke prependNode

5. **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.

6. **Function:** examineLocation
   - This function prints all treasures found in a specified location
   - A list_iterator has been provided.
   - This list_iterator has been initialized.
   - Iterate—i.e., loop—through the linked list.
     - If an item location, list_iterator->data->location, matches location.
       - Print the found ‘index’ and treasure name
## Sample Output:

Enter the input filename: 810-input.txt

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

Treasures Found: 6

Remove which Node: 1

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TA-Office</td>
<td>NES</td>
</tr>
<tr>
<td>2</td>
<td>Dragas</td>
<td>SNES</td>
</tr>
<tr>
<td>3</td>
<td>Dragas</td>
<td>Wii</td>
</tr>
<tr>
<td>5</td>
<td>Spong</td>
<td>Nothing</td>
</tr>
</tbody>
</table>

Treasures Found: 5

New Treasures

What did you find? : 3DS
Where was it found? : Dragas
Insert at head or tail? <H/T>: T

What did you find? : Lemon & Ginger Tea
Where was it found? : TA-Office
Insert at head or tail? <H/T>: H

What did you find? : Pancakes
Where was it found? : IHOP
Insert at head or tail? <H/T>: T

What did you find? : Linked Lists
Where was it found? : Dragas
Insert at head or tail? <H/T>: T

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TA-Office</td>
<td>Lemon &amp; Ginger Tea</td>
</tr>
<tr>
<td>2</td>
<td>TA-Office</td>
<td>NES</td>
</tr>
<tr>
<td>3</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>
<tr>
<td>8</td>
<td>IHOP</td>
<td>Pancakes</td>
</tr>
<tr>
<td>9</td>
<td>Dragas</td>
<td>Linked Lists</td>
</tr>
</tbody>
</table>

Treasures Found: 9

Examine which location? : Dragas

<table>
<thead>
<tr>
<th>#</th>
<th>Location</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NES</td>
<td>SNES</td>
</tr>
<tr>
<td>3</td>
<td>Wii</td>
<td>SNES</td>
</tr>
<tr>
<td>4</td>
<td>3DS</td>
<td>SNES</td>
</tr>
<tr>
<td>5</td>
<td>Linked Lists</td>
<td>SNES</td>
</tr>
</tbody>
</table>

Total: 5