Data Structures Name -
Ch. 3 Practice Test

The Ch. 3 Test will consist of true/false, multiple choice, fill-in-the-blank, & fill-in-the-missing-code problems. You will also have to write one or more complete member functions for the author's `LinkedList` class and you will have to write one or more complete free functions using struct definitions provided on the test. Be sure to study the code in the implementation of the author's `LinkedList` class, especially the insert and remove member functions.

Do the following sample problems on lined paper in order to be fully prepared for the Ch. 3 Test.

1. Using the author's `LinkedList` class, write a free function named `addZeroes` that adds one node with the data value of zero in front of each existing node that contains a value greater than 100.

2. Write a member function `switchFirstAndLast` that could be added to the author's `LinkedList` class and that interchanges the first and last nodes of a linked list. Private data of the list must be updated appropriately.

3. Given the following struct definition, write a free function named `sumPositives` that returns the sum of all of the positive integers stored in a linked list of integers. You are not working with the author's `LinkedList` class in this problem.

```
struct node
{
   int data;
   node * next;
};


int sumPositives(node * head)
// precondition: head points to a node in a
//       linked list of nodes or head points to null if the list
//       is empty
// postcondition: the sum of all of the positive data members
//       will be returned. If there are no nodes, the value 0
//       should be returned.
```

4. Write the function definition for the author's `LinkedList` class `insert` member function.

```
template <class E>
void LinkedList<E>::insert(const E &item)
```