Chapter 17: Linked Lists:
Inserting a Node

Starting Out With C++
From Control Structures through Objects
EIGHTH EDITION
Tony Gaddis

© Pearson Education Limited 2015
Inserting a Node into a Linked List

- Used to maintain a linked list in order
- Requires two pointers to traverse the list:
  - pointer to locate the node with data value greater than that of node to be inserted
  - pointer to 'trail behind' one node, to point to node before point of insertion
- New node is inserted between the nodes pointed at by these pointers
Inserting a Node into a Linked List

New node created, correct position located
Inserting a Node into a Linked List

New node inserted in order in the linked list
void NumberList::insertNode(double num) {
    ListNode *newNode; // A new node
    ListNode *nodePtr; // To traverse the list
    ListNode *previousNode = nullptr; // The previous node

    // Allocate a new node and store num there.
    newNode = new ListNode;
    newNode->value = num;

    // If there are no nodes in the list
    // make newNode the first node
    if (!head)
    {
        head = newNode;
        newNode->next = nullptr;
    }
    else // Otherwise, insert newNode
    {
        // Position nodePtr at the head of list.
        nodePtr = head;
    }
}
// Initialize previousNode to nullptr.
previousNode = nullptr;

// Skip all nodes whose value is less than num.
while (nodePtr != nullptr && nodePtr->value < num) {
    previousNode = nodePtr;
    nodePtr = nodePtr->next;
}

// If the new node is to be the 1st in the list, insert it before all other nodes.
if (previousNode == nullptr) {
    head = newNode;
    newNode->next = nodePtr;
} else // Otherwise insert after the previous node.
{
    previousNode->next = newNode;
    newNode->next = nodePtr;
}
Program 17-3

```cpp
// This program demonstrates the insertNode member function.
#include <iostream>
#include "NumberList.h"
using namespace std;

int main()
{
    // Define a NumberList object.
    NumberList list;

    // Build the list with some values.
    list.appendNode(2.5);
    list.appendNode(7.9);
    list.appendNode(12.6);

    // Insert a node in the middle of the list.
    list.insertNode(10.5);

    // Display the list
    list.displayList();
    return 0;
}
```

Program Output

2.5
7.9
10.5
12.6

How does the code look like to display list?
Inserting Element - Questions

- Appending a node we only need `nodePtr`
- Why do we need the `previousNode` pointer to insert a node?
Inserting a Node into a Linked List

New node created, correct position located
Why do we check whether `newNode` will be the first node of the list?
Inserting a Node into a Linked List

previousNode

null

nodePtr

New node created, correct position located
Inserting a Node into a Linked List

previouNode: null

nodePtr

list head

newNode

New node created, correct position located
How can we insert a node without using the previousPointer?
Inserting a Node into a Linked List

How should we change the code to insert newNode without using previousNode?

New node created, correct position located