Subject: Trouble with pointers.
Posted by snap2000 on Sat, 25 Aug 2007 03:04:41 GMT
View Forum Message <> Reply to Message

I was trying to make a LinkedList class and it seemed to be going pretty well, but I've run into an annoying stumbling block:

In file included from C:\Apps\Classes\LinkedList\main.cpp:4:
C:/Apps/Classes/LinkedList/LinkedList.cpp: In member function `LinkedList::Node&
LinkedList::Push()':
C:/Apps/Classes/LinkedList/LinkedList.cpp:56: error: invalid initialization of reference of type
'LinkedList::Node&' from expression of type 'LinkedList::Node*'
LinkedList.cpp
C:\Apps\Classes\LinkedList\LinkedList.cpp: In member function `LinkedList::Node&
LinkedList::Push()':
C:\Apps\Classes\LinkedList\LinkedList.cpp:56: error: invalid initialization of reference of type
'LinkedList::Node&' from expression of type 'LinkedList::Node*'

Here is the code that I have:

LinkedList.h

```
#ifndef _LinkedList_LinkedList_h_
#define _LinkedList_LinkedList_h_

class LinkedList {
public:
 struct Node {
  int  uid;   // Node identifier
  Node *prev;   // Pointer to the previous node
  Node *next;   // Pointer to next node
 };
private:
 Node *first;   // Pointer to first item
 Node *last;   // Pointer to last item
 Node *current;  // Pointer to current item (iterator)
 int  count;   // Number of items in list
 int  uid;   // Unique identifier for nodes

public:
 typedef LinkedList CLASSNAME;

 LinkedList();   // Constructor
 ~LinkedList();   // Deconstructor

 Node& First()   { return *first; };   // Returns first item
 Node& Last()   { return *last; };   // Returns last item
 Node& Current()  { return *current; };  // Returns current item (iterator)
 int  Count()   { return count; };   // Returns item count
```

```
 Node& Iterate();  // Returns each item in the list successively
 void Reset();  // Resets iterator to first item

 Node& Push();   // Adds item to back of list
 Node& UnShift();  // Adds item to front of list
 Node& Pop();   // Removes item from back of list
 Node& Shift();  // Removes item from front of list

 void Remove();
};


#endif
```

LinkedList.cpp:

```
#include <iostream>

#include "LinkedList.h"

LinkedList::LinkedList() {
 first = NULL;
 last = NULL;
 count = 0;
 uid = 0;
}

LinkedList::~LinkedList() {
 Node *temp;
 /*
 while( ( temp = Iterate() ) != NULL ) {
  delete temp;
 }
 */
}

LinkedList::Node& LinkedList::Iterate() {
 /*
 Node *temp = current;
 if( current != NULL )
  current = current->next;
 else
  Reset();
 return *temp;
 */
}
```

```
void LinkedList::Reset() {
 current = first;
}

LinkedList::Node& LinkedList::Push() {
 Node *temp;

 // Set new node attributes
 temp = new Node;
 temp->uid = ++uid;
 temp->prev = last;
 temp->next = NULL;

 if( last != NULL )
  last->next = temp;  // Link previous item to new node

 last = temp;
 if( !count )    // If list is empty, new node is also first
  first = temp;

 count++;

 return temp;
}

LinkedList::Node& LinkedList::UnShift() {

 count++;
}

LinkedList::Node& LinkedList::Shift() {
 count--;
}

LinkedList::Node& LinkedList::Pop() {
 count--;
}

void LinkedList::Remove() {
 count--;
}
```

The error only goes away when I return *temp instead of just temp. However, if I do this and uncomment the contents of the destructor, I get the following error instead:

```
In file included from C:\Apps\Classes\LinkedList\main.cpp:4:
C:/Apps/Classes/LinkedList/LinkedList.cpp: In destructor `LinkedList::~LinkedList()':
```

C:/Apps/Classes/LinkedList/LinkedList.cpp:15: error: cannot convert `LinkedList::Node' to `LinkedList::Node*' in assignment
LinkedList.cpp
C:\Apps\Classes\LinkedList\LinkedList.cpp: In destructor `LinkedList::~LinkedList()':
C:\Apps\Classes\LinkedList\LinkedList.cpp:15: error: cannot convert `LinkedList::Node' to `LinkedList::Node*' in assignment
LinkedList: 2 file(s) built in (0:00.79), 398 msecs / file, duration = 828 msecs

I thought I understood pointers pretty well, but I just don't know what's going on. Help please?