

I have a struct with the following structure:

```
struct LLxml{  
  
    String tag;  
    Array<LLxml> childs;  
    VectorMap<String, String> attribs;  
    String inner_text;  
  
};
```

The point of this struct is to store the results of an XmlNode analysis in hierarchal fashion. The tag name is stored in tag, the tags that are children of this tag are stored in childs, the attributes of the tag are stored in attribs, and (if there is any) free-standing text in a tag is stored in inner\_text.

Now, my problem is that when I'm analyzing the data stored into the struct (which is essentially a pseudo-linked list), as I'm traversing through the children I need to copy the parent values. So that, if I had a variable root, representing the top level tag, I get something like:

```
LLxml root;  
LLxml temp = root; // to store the root  
  
for(int i = 0; ...){  
    root = root.childs[i];  
  
    ...  
}  
  
root = temp;
```

Now, as some might see, when I try to execute this, I get all kinds of problems. If I don't get "Broken Pick Semantics" problems, the value that fills root is incorrect.

I suppose this is an issue dealing with how variables and such are copied, but I'm not really sure what to do.

Any help is greatly appreciated.

---

---

Subject: Re: Real newbie question about copying values  
Posted by [gonzofish](#) on Wed, 06 Feb 2008 19:36:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Like I said, newbie question. I solved my problem through the use of a pointer.

Thanks to anyone who gave the problem a look.

---