Subject: c- constant multi char and string variable
Posted by oamohamed@crimson.ua.edu on Sat, 03 Sep 2016 05:59:02 GMT
View Forum Message <> Reply to Message

About the undeclared variable, I am having a little trouble dealing with it, whether I treated it as a constant char * sort of a token or as a C string type. I keep getting a segmentation fault on the running time.

Subject: Re: c- constant multi char and string variable
Posted by luth on Sat, 03 Sep 2016 14:26:02 GMT
View Forum Message <> Reply to Message

I do not use the keyword const at all in my C programs. My view is that if one follows good coding practices, these types of compile "hints" are unnecessary.

You should not be using multiple character constants in this program. Also if x is declared char *, then *x has type char.

You may post the error message and the line of code that caused it (and the preceding line if it's important).

Subject: Re: c- constant multi char and string variable
Posted by oamohamed@crimson.ua.edu on Mon, 05 Sep 2016 03:29:05 GMT
View Forum Message <> Reply to Message

I am not sure if checking the first char on the "var" input will be sufficient for this assignment, I also have a question about the look up structure "tree", is the tree has to be search tree? or a binary tree will be acceptable?.

Subject: Re: c- constant multi char and string variable
Posted by luth on Mon, 05 Sep 2016 13:14:54 GMT
View Forum Message <> Reply to Message

"v" is a valid variable name, as is "va" and "vary".

Binary search tree, as the spec states.

Subject: Re: c- constant multi char and string variable
Posted by oamohamed@crimson.ua.edu on Sun, 11 Sep 2016 15:29:51 GMT
View Forum Message <> Reply to Message

if x declare char *x, will a comparison of this type be valid in bst update; ex
Subject: Re: c- constant multi char and string variable
Posted by lusth on Mon, 12 Sep 2016 10:59:22 GMT
View Forum Message <> Reply to Message

No, you have to use strcmp. I posted about how to use strcmp in a bst in another thread.