

Re: for and arrays

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2008-07/msg03156.html

- *From:* Richard <rgrdev@xxxxxxxxxx>
 - *Date:* Mon, 28 Jul 2008 03:46:49 +0200
-

Ben Bacarisse <ben.usenet@xxxxxxxxxx> writes:

"Default User" <defaultuserbr@xxxxxxxxxx> writes:

<snip>

Riley is a troll.

For the record, I don't use the term myself. It is too easy to throw around and too hard to decide in all the cases where it matters.

We've been over this.

If you mean by this that you've given me your opinion before, then I take your word for it. If you mean that most people here are of that

"over this" – he has given his instructions.....

opinion, then I agree, and that is in part why I posted. If anyone else had chosen to point out how a debugger is unlikely to help in this case, I would not have felt the need.

The point is that it would. You might not see it. But you are looking from a different, possibly equally valid, angle. I have spent a lot of time with trainees and I have seen people almost as stuck as Bill. Best way is to get them into the memory and SEE things as the program steps. To see the string. To see the chars. To see the end of string marker.

And if you think this advice is trolling then I am surprised.

Re: for and arrays

I have found "Plain Richard" to be someone that I do not want to argue with. He does not try to understand other people's positions and will, in the end, resort to claiming that he just does not believe his opponents.

In some cases I take that stance. For example when someone assures me they can debug 50,000 lines of foreign c code quicker from a printout than from using an industry strength debugger. Those kind of things that only get spouted in clc. it is nonsense. Can someone in the world? I have no doubt someone can. Would we recommend that to someone like Bill because some Indian Guru can? No we would not. Would we guess that 99% can because Heathfield once spotted a bug on page one 10 years ago? No we would not. We would advise sensible procedures to approach the problem.

That does not mean, to me, that everything he writes should go unchallenged. Sometimes I want to offer an opposing view but you will see that I am quite happy to let him have the last word. He has explained his position and I have clarified mine.

My position here is that its the old seed versus crops thing with Bill. Get him to teach himself the basic types. Feeling them with a debugger is a great way. He seems to be as clueless as 6 months ago and need to get his hands dirty. Pasting in solutions for him is not helping him – if anything it is hindering him since he then mistakenly thinks he knows why something is working. He doesn't. He needs to follow the tutorial and learn how to examine his own programs for now. More complicated things can follow.

Of course you (and others) are free to tell me whose posts you think I should ignore. I take that advice seriously (but not as gospel). For some reason (which I really don't understand) I don't feel I want to offer such advice myself.

Bwian does it a lot. He has been called on it a lot. I have no idea why he feels the need to tell others who or what to listen to. Frankly I think he sees himself as Heathfield's right hand man or Rottweiler. I'm not sure which.

.