

Re: Letter to US Sen. Byron Dorgan re unpaid overtime

Source: <http://coding.derkeiler.com/Archive/General/comp.programming/2003-12/2036.html>

From: Randy Howard (*randy.howard_at_FOOmegapathdslBAR.net*)

Date: 12/23/03

Date: Tue, 23 Dec 2003 13:36:08 -0600

In article <Xns945A9F07CE465newspubwuggyorg@217.32.252.50>, newspub2@wuggyNOCAPS.org says...

> > *He prides himself on "knowing" that using strlen in the for loop*
> > *terminating condition is bad...but the knowledge cannot be*
> > *generalized, outside C.*
>
> *This is nonsense: the generalisation, and why it is bad, isn't about*
> *using strlen but using an $O(n*n)$ algorithm when there is a trivial $O(n)$*
> *algorithm.*

You have managed to boil this entire thing down very succinctly. Nilges could learn a lot from that. The fact is, he doesn't understand why it matters. His statements on this topic imply that "as long as I don't let n get too big, it'll never matter". In a certain sense, that holds true, the problem being that the developer can never guarantee what the end user will do 100% of the time. More importantly, those that read, modify or worst of all in his case, learn from such code have to suffer through this crap, in some cases without understanding why it is a problem.

I suspect one potential cause of this is that Nilges cannot imagine any of his work output being reused, as such there is no chance that a routine with $O(n*n*n)$ complexity will actually be used by anyone else [come on, we know he's done that at least once, the code he's proud of is $n*n$, despite not needing to be] . Within those guidelines, he won't ever see the problem rear its head. However, anyone that reads his stuff and believes it will be programming in mud hits in third world countries soon, which may in fact be his goal. Karmic retribution or some such nonsense.

> *I don't care what language is being used, if I see a colleague or*
> *even a student of mine using an $O(n*n)$ algorithm when it takes no*
> *more programmer effort to use the trivial $O(n)$ algorithm, I'll tell*
> *them so.*

Yes. And I've never met anyone other than this sociologist who

would argue against using it and would not gladly accept the knowledge imparted. The inability to learn from his mistakes is by far the most damning aspect of this whole spew-fest.

> *You chose to make an $O(n*n)$ algorithm out of an $O(n)$ algorithm by /not/ using knowledge trivially available to you as a programmer which isn't available (or trivially available) to the optimiser. You then argued about the time complexity of the operation. This shows at best that you have a misplaced trust in the tools at hand, or at worst that you shouldn't be programming.*

I think the latter is more apropos in this case.

> *Back to the front though. Obviously then, this Algol-60 for loop is also "deviant and passive aggressive":*

:~)

> *for NewGuess := Improve(OldGuess)*
> *while abs(NewGuess - OldGuess) > 0.0001*
> *do OldGuess := NewGuess;*
>
> *It wasn't, of course, designed by Ritchie unless he secretly snuck into the committee room without anyone noticing.*

It was a Communist conspiracy of course. Can you imagine what Nilges could have been had he worked for Joseph McCarthy as an "intern"? s/blue dress/blue suit/

shudder

BTW, EDG, can you explain why "john nash" and "nilges" appear in zero internet hits not authored by you? Did you just make this up? Will anyone admit to the association other than you?

Hmmm, maybe YOU are the reason he was seeing things. I can certainly understand how an escape route to get away from your BS could be a good thing.

> *you write all the trivial $O(n)$ algorithms as $O(n*n)$ right from the start, 'optimising' the code is going to be a rather painfull experience.*

Only if you actually try it. :~)

--
Randy Howard _o
2reply remove FOOBAR \<,
 ()/ ()

SCO Spam-magnet: postmaster@sco.com