

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

**Source:** <http://coding.derkeiler.com/Archive/General/comp.programming/2003-12/1984.html>

---

**From:** Richard Heathfield (*dontmail\_at\_address.co.uk.invalid*)

**Date:** 12/23/03

Date: Tue, 23 Dec 2003 07:56:56 +0000 (UTC)

Edward G. Nilges wrote:

> *Richard Heathfield wrote...*

>> *Edward G. Nilges wrote:*

>>

>> > *Instead, you judge*

>> > *people by fabricated metrics which inappropriately foreground*

>> > *transient issues, like a poor head-hunter.*

>>

>> *No, I just find it hard to believe anyone could be so stupid as not to*

>> *cache processor-intensive calls.*

>

> *Yeah, while letting the system as a whole go to hell.*

Um, what? Are you saying that maximising the time it takes to perform your task somehow increases overall system performance? I mean, I wouldn't want to put words in your mouth or anything, so I thought I'd better check. After all, this is even sillier than your O(1) strlen.

> *I don't think*

> *you know how minimal the cacheing under consideration would be in*

> *effect.*

I've showed that the effect is vast.

>> > *Richard, I was frankly shocked to learn that you are unclear on the*

>> > *very idea of a runtime instruction that would scan a string for a Nul*

>> > *character,*

>>

>> *Oh, I'm not unclear about it AT ALL. I have posted results elsethread*

>> *that show quite conclusively that you're talking through your nose (for*

>> *the benefit of anyone still naive enough to think you have the slightest*

>> *idea what you're talking about).*

>>

> *The results are always posted somewhere else, aren't they,*

No, they're always posted where they're posted. Learn To Read. Learn To Use

Your Newsreader.

But JUST FOR YOU, I'll repeat them right here. The following timings are for a single line of input:

Line length	My way (s)	Your way (s)	Your way / My Way
256	0.010	0.010	1.0
512	0.010	0.010	1.0
1024	0.010	0.013	1.3
2048	0.010	0.025	2.5
4096	0.011	0.068	6.1
8192	0.012	0.240	20.0
16384	0.014	0.945	67.5
32768	0.019	5.221	274.7
65536	0.035	20.971	599.1
131072	0.047	83.508	1776.7

Note that, after a certain period of grace, "my way" starts to exhibit classic O(n) behaviour. Now look at yours. Sheesh, man.

> *because you*  
> *are not in this conversation to conduct a dialog. Instead you are*  
> *playing to a sort of peanut gallery.*

Actually, I am frantically /trying/ to conduct a dialog, but it's difficult when dealing with someone as brainless and as incapable of reading as you make yourself appear to be.

[Just to be clear, the following is EGN]

>> >> > *Also, this is an ugly piece of code. It increments the line counter*  
>> >> > *unnecessarily since rc is set to a value, then thisline is*  
>> >> > *incremented (for some strange reason, in a nonstandard fashion, in a*  
>> >> > *preincrement), then the condition is evaluated.*  
>>  
> *I did not say that the condition is evaluated after the increment,*

A flat lie. That's precisely what you said.

> *since I started with the assignment to the rc, in the middle, and then*  
> *narrated what happens. I know the sequence of evaluations, but in*  
> *trying to give the false impression that I don't to add to your absurd*  
> *charge sheet, you gave the impression you don't.*

People who can read know precisely what you said, and they can always read it again if they wish.

> *That's because you've entered this dialog dishonestly and as a bully.*

No, I entered this dialog to correct your mistakes.

> *Your bad attitude causes misunderstanding at warp speed,*

No, your ignorance causes mistakes at warp speed. I have not misunderstood anything in this thread.

> *and, it is an*  
> *example of what I was talking about when I pointed out that*  
> *programmers can no longer conduct effective walkthroughs.*

Actually, programmers /can/ conduct effective walkthroughs, and regularly do so, \*even though\* you think they can't. Isn't that amazing?

<nonsense snipped – if I answered every point, I'd be here till Christmas.  
Brief summary: I disagree with the stuff I snipped>

```
>>> for (intIndex1 = 0; intIndex1 < strlen(strInstring); ++intIndex) { ...
>>> }
>>>
>>> and I REALLY hope you know that the iteration is performed AFTER the
>>> iteration.
>>>
>>> Typical Nilges – the iteration is performed after the iteration, is it?
>>> Well, that's just wonderful. Here on Earth, we prefer common sense.
>
> No, just a typo.
```

Ah, okay, I'll accept that. It's hard to tell with you. After all, you can't even read the stuff I write correctly; God help me if /I/ make a typo. You'd jump on it like a starving stoat.

> *I don't want to waste too much time with you, and*  
> *your bullying upsets me,*

But I don't bully. In your case, I freely admit that your steady stream of abusive, offensive filth gives me less incentive than usual to remain my normal mild-mannered self; if you don't want your errors to be treated harshly, stop abusing the newsgroup. If you keep abusing the newsgroup, expect your errors to be treated harshly.

> *so I make errors. Stop the insanity and there*  
> *will be fewer typos...on both sides.*

(I'm not a qualified psychiatrist, so I carefully point out that, in the following sentence, I use the word "insanity" in a loose, colloquial sense.) The only insanity here is yours.

> *I see, when you type in code you can make errors, but I can't.*

Sure you can. You do it all the time. Anyway, the typo was the least of your problems. Learn to use a debugger, and if you must put in printf's, don't put them in stupid places. Those points are far more important.

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

> *Of course, C requires some silly symbol in front of d.*

No, not just any old silly symbol. It has to be a particular silly symbol.

> *You are playing games because you had dominated this ng*  
> *inappropriately with commercial promotion of the C language when I*  
> *posted the popular thread on Steve McConnell's book.*

No, I didn't. I have never dominated this newsgroup, nor have I ever commercially promoted the C language here. I have recommended C to some individuals. I have recommended Java to at least one. I have even recommended Pascal to – well, I don't know how many, but I would think it's more than one. BTW, LOL @ "the popular thread". That was funny.

> *You are*  
> *determined to drive me away which is common practice in ngs*

No, I'm determined to oppose off–topic pollution of the group. If you want to post topical stuff here, fire away. That'll even be amusing, given your natural incompetence in the field of programming.

>> > *At the point of early exit on fail, the index in thisline will*  
>> > *still contain a zero based value because ++thisline doesn't have the*  
>> > *effect you seem to think it does.*  
>>  
>> *Nonsense. Here's proof:*  
>>  
<proof snipped for brevity>  
>  
> *This code will produce a one based index properly.*

Ah, finally you admit you're wrong. Good grief. Put the flags out.

> *It goes through a*  
> *contortion to work*

No, the correct behaviour happens to fall out.

> *because you're not man enough :-)* *to do the right*  
> *thing and put a break where it belongs.*

Don't be so childish. Of course I can put a break there if I choose. I choose not to.

> *The contortion is that you set a return code to show what is a break*  
> *condition,*

What is a "break condition"? As far as I'm concerned, I want to keep looping until I've done all the lines or until something goes wrong. That is the loop termination condition, so that's what I test. This is obvious to me.

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

<something stupid about fetishism snipped>

- > *A prohibition ("a loop must have only one exit"), an urban legend,*
- > *replaces science in a dialectic of enlightenment, and this causes you*
- > *to create YET ANOTHER ugly "temp" variable, rc. Each one of these busy*
- > *little sods create a possible point of failure but this is no*
- > *nevermind to you: you've obeyed a rule.*

No, I've made my code easier for me to read, write, and understand. Programming is difficult (as you must surely appreciate, being as hopeless at it as you apparently are), so I like to make it easier.

- > *Why not just break and in the report, Code What You Mean by explicitly*
- > *incrementing the variable?*
- >
- > *for ( intIndex1 = 0; intIndex1 < strlen(strInstring); intIndex1++ )*
- > *if ( rand() % 11 == 0 ) break;*
- > *printf( "At character %d of &d\n", intIndex+1, strlen(strInstring) );*

Like that, you mean? Further evidence of your incompetence. (Note that you screwed up the printf again. You're good at that. Hint 1: & is not %, but we'll let that off as a typo, okay? Hint 2: strlen does not return int.)

- >
- > *Cudgel thy brains over this code:*
- >
- > *(1) It uses Hungarian notation because Hungarian notation makes it*
- > *more readable. End of story.*

conjIf adjHungarian nounnotation vrbmakes pronit adjmore adjreadable, why don't you use it in English? Never heard such nonsense.

- > *(2) The use of strlen is only micro-inefficient*

That depends on the size of the data set, which often cannot be known in advance.

- > *EVEN IF the compiler*
- > *does not optimize the strlen to a single machine language instruction,*

Oh, nobody's disputing that the compiler can do that if such an instruction is available. What we're saying is that the usage of a single machine instruction doesn't imply a constant running time.

- > *and this is because if the application is properly designed, most*
- > *strings will be in the range 1...1000 characters. The developer will*
- > *be MORE concerned with ensuring that long strings are not passed in*
- > *the first place because of the global resource consumption.*

1000 characters? That seems to me to be a very arbitrary, and artificially small, limit. Let's have a look at what a certain Mr Nilges has to say

about artificially small limits:

"The problem is that to do common standard things in C you have to enter the minefield. It may have been an error in traditional VB to make a string a basic data type because of the garbage collection problem, but at a minimum it did not exclude the Nul character and, alternatively, did not after VB-3 impose an artificially small limit on string length." – Edward Nilges, 26 September 2002.

Hmmm. Well, my solution is good for megabytes. Yours isn't even good for kilobytes.

- > *For example, and in a compiler, the developer with a vision larger*
- > *than a few lines of code, will know that the above code SHOULD NOT BE*
- > *USED,*

Then why bother writing it?

- > *(3) The use of strlen follows Kernighan's rule in The Elements of*
- > *Programming Style: code what you mean.*

Nobody said "don't use strlen". But counting characters has a cost. When I use strlen to calculate the number of characters in a string, I generally mean to do it once, so I only call strlen once. Why call it more often? It's silly, and not what I mean – therefore I don't write it like that. And anyway, the cost of interpreting Kernighan's advice the way you do is unacceptably high.

- > *If your C compiler generates a*
- > *linear search for the end of the string, get an optimizing compiler or*
- > *abandon C.*

On the other hand, you could just write sensible code.

- >
- > *(4) Last but not least, the break visually makes it plain that the*
- > *loop anticipates an early exit condition. Your code hides this fact.*

No, it demonstrates, clearly and unambiguously in the very loop condition, under exactly which circumstances the loop will stop executing. You can't miss it.

<big old snip>

- > *I am relieved that you know the order of evaluation in a for*
- > *statement. However, you don't see that the for statement hides an*
- > *important fact, and this is that it evaluates variables "by reference"*
- > *each time through.*

That is not hidden at all. It's out there in the open, proclaimed by an International Standard.

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

- > *It cannot be optimized to put the limit in a fast*
- > *register and this "feature" belongs in a distinct statement.*

Yes, it can be so optimised, in just the way I have shown.

<snip>

--

Richard Heathfield : [binary@eton.powernet.co.uk](mailto:binary@eton.powernet.co.uk)

"Usenet is a strange place." - Dennis M Ritchie, 29 July 1999.

C FAQ: <http://www.eskimo.com/~scs/C-faq/top.html>

K&R answers, C books, etc: <http://users.powernet.co.uk/eton>