

# Re: pure functions

---

*Source:* [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2005-06/msg01420.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2005-06/msg01420.html)

---

- *From:* "Malcolm" <[regnizar@xxxxxxxxxxxxxxxx](mailto:regnizar@xxxxxxxxxxxxxxxx)>
  - *Date:* Fri, 10 Jun 2005 21:30:57 +0000 (UTC)
- 

"Keith Thompson" <[kst-u@xxxxxxx](mailto:kst-u@xxxxxxx)> wrote

>

> Discussion of pure functions within the context of standard C, and the  
> opportunities they can provide for potential optimizations, is  
> topical. Discussion of implementation-specific features to support  
> pure functions are off-topic (<OT>I'll take the liberty of mentioning  
> gcc's `__attribute__((pure))`, but I won't go into detail</OT>).  
> Advocacy of such features for future standards is probably more  
> topical in `comp.std.c`, but I see no real problem with bringing it up  
> here.

>

Discussion about how to use C effectively, so that we can enjoy the benefits  
of pure functions despite the fact that the language provides no way to  
enforce lack of side-effects, is also topical.

For instance you could have a convention that pure functions always take  
single-letter parameters, non-pure functions always take at least one  
parameter with more than one letter.

.

---

- *Follow-Ups:*
  - ◆ **[Re: pure functions](#)**
    - ◇ *From:* Chris Coughton
- *References:*
  - ◆ **[pure functions](#)**
    - ◇ *From:* junky\_fellow
  - ◆ **[Re: pure functions](#)**
    - ◇ *From:* Malcolm
  - ◆ **[Re: pure functions](#)**
    - ◇ *From:* pete
  - ◆ **[Re: pure functions](#)**
    - ◇ *From:* Keith Thompson
- Prev by Date: **[Re: sizeof\(\) style question](#)**

Re: pure functions

- Next by Date: ***Re: The status of C***
- Previous by thread: ***Re: pure functions***
- Next by thread: ***Re: pure functions***
- Index(es):
  - ◆ ***Date***
  - ◆ ***Thread***