Re: C is too old? opinions?

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2006-07/msg01984.html

- *From*: Sjouke Burry <burrynulnulfour@xxxxxxxxxxxxxxxxxx>
- *Date*: Fri, 14 Jul 2006 00:01:09 +0200

W Marsh wrote:

> On Thu, 13 Jul 2006 11:53:06 −0700, "Dann Corbit" <dcorbit@xxxxxxxxx>
> wrote:
>
> > "W Marsh" <wayne.marsh@xxxxxxxxx> wrote in message
> > news:oc5db2d2fqj4jkht7ihl2pr45javrb8ee5@xxxxxxxxxx
> >
> > > On Thu, 13 Jul 2006 11:45:02 −0700, "Dann Corbit"
> > > <dcorbit@xxxxxxxxx>
> > > wrote:
> > >
> > > > Languages do not age the way that people
> > > > think they do.
> > > > COBOL and BASIC are alive and well.
> > > > C will never go away.
> > > > For writing simple filter programs, it is a
> > > > very good alterative.
> > > > Languages like C# and Java have to go
> > > > garbage collection and hence are
> > > > unsuitable for real−time stuff.
> > > > Can you imagine .NET installed on a toaster
> > > > IC to pop your toast up in the
> > > > morning? I can't.
> > >
> > > Java is used extensively on embedded devices. You're
> > > making yourself
> > > look a bit silly, there.
> >
> > How does your response relate to my post?

> Is Java used for real−time systems (if so, they should fire the product implementors for incompetence).
>
> Does Java require .NET?
>
> I think you have a reading comprehension problem.

> You were lumping Java and .NET together, suggesting that they were inadequate in real−time systems for the same reason. I assumed you must have been talking about embedded stuff as well because of your vague reasoning and toaster example*. In other words, you seemed confused. I made a logical leap as required.
>
> Go on then − I would like to know how garbage collection affects real−time systems. A good reason, showing thought and reason. In other words, show us that you're not just full of shit.
>
> Bear in mind that garbage collection shouldn't affect the behaviour of a system, only its constraints.
>
> * Exactly how precise do you think a toaster timer needs to be?

How precise does a feedback loop in control
need to be,when the loop goes round 1000
times per second? Even malloc/free can be
to much there.
With LISP as an example, calculating a fractal,
it stopped every 2 seconds,to do garbage collection
for 1 or two seconds.
Things just depend on how real time your
program is,and hoe short your time tick is.
If short enough,any housekeeping job can be to much,
in which case I am glad to have an old
C compiler,without those nice new features.
.