The routines have been structured like Mr. Bauer said in his first
reply. I would like to hear your comments about how we can improve the
organization of our program.

One of my confusion has not been resolved yet. For example on a modern
embedded C compiler, throughout the debugging process if I make a
change on the source code and save the file, and try to observe the
same function by executing a command or function, the compiler warns
me to re-compile the code. While doing this, it only compiles the
recently changed file (I mean as far as I understand the compiler or
linker attaches a file stamp or flag them, so next time it doesn't re-
compile unchanged files. This means a significant compiling speed
improvement and this is something I haven't observed in IDL.

On Nov 6, 11:34 am, David Fanning <n...@dfanning.com> wrote:
> gsever writes:
>> Once again, in the same IDL session, when I first launch the program
>> 168 routines are compiled, after I make an addition the source code of
>> one file, and re-run the program all 168 routines are re-compiled
>> again. This is controversial to Reimer Bauer's yes answer.
>
> Yes, it suggests a poorly written IDL program, I'm afraid. :-(
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")