

Re: function pointers

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2008-01/msg01140.html

- *From:* jacob navia <jacob@xxxxxxxxxxx>
 - *Date:* Wed, 09 Jan 2008 01:22:56 +0100
-

Ian Collins wrote:

CBFalconer wrote:

Ian Collins wrote:

cr88192 wrote:

.... snip ...

on linux, this option works better though,
since linux crashes
give a general debugger–usable core dump
(sadly absent on
windows...).

I pity those poor windows developers :)

Why? I get all that ability. Of course, I am using gcc under
DJGPP on winders.

You prove my point, jumping through hoops to get a useful environment.

I am programming in debian, I have a customer under linux that
wants a server application.

I installed debian (latest).

To boot, I start with a menu of the "grub" booting utility.

I point the cursor at the first line, then I press "e", then
"e" again, then I change (hd1,0) to (hd0,0). Then I
scroll to the second line and press "b". Then I am able to boot.

This is due to a bug in debian that mixed up the 3 disks in my machine.

Re: function pointers

After this, I boot only when I do not do a restart. Debian needs a cold restart to boot. Any warm restart (ctrl+alt+del) doesn't work and the boot process hangs after a few seconds.

When I arrive at the desktop, I type my name and see a thing called "gnome". It is a great desktop. Sound doesn't work, nor my music collection, anyway who cares. The Desktop shows some icons about the different disks. There is no way to hide those useless icons. The desktop shows very often the cdrom icon on top of another icon. Then you have to move it each time you insert a cdrom to access the cdrom contents.

Debian doesn't like that people go away. When I go away it will start a desktop screen saver program. I changed the screen saver to a blank screen, then, I discovered that the screen saver doesn't come back, i.e. you can type like crazy at the keyboard but the screen REMAINS blank.

Great. I lost everything and now I have the screen saver time delay at 10 000 minutes...

There is no IDE, nor any modern development system. I use vi, gdb, and the compiler, exactly like in 1986 when I first started using Unix.

The different "IDE"s are all completely crap. No go to definition, no browse by function, no nothing. They are barely able to start the compiler.

There is only one debugger: gdb. All other "debuggers" are just software on top of that, but since they did not integrate the debugger source code into the IDE they offer NO extra functionality but they offer a layer of BUGS that is quite incredible.

Printer doesn't work, but this is no great pain since I can always reboot and print under windows.

There is no obvious way to write to a DVD to make a backup. I looked at some "HOWTOS" (that goes as documentation this days), but they were outdated. Obviously, for Unix gurus I am just a looser that doesn't know how to use the system, and that is right. I did not want to learn how to write obscure commands to make the sound work, to download software from the internet to make the cd writer work, to study CUPS and make the printer work, to study GRUB and see why my changes to the configuration file are overwritten...

Yes, I *could* learn all that if I wanted to... I learned enough of GRUB to find out that the grub system mixed the hard disks and that I had to replace hd1 by hd0 using the built in editor (took me 2 hours) of googling, reading, etc.

Re: function pointers

That was a FATAL error and I HAD to fix it. But I can live without sound, printer, screen saver, and I can develop using vi+gcc+gdb. It is horrible but if it is well paid I do not give a damm. I have programmed in worst environment than this one.

At least, vi works, and I do not have to use ed

:~)

I am a linux user since it came out with the first slackwares... AND I will go on using it, mind you. As long as there are customers under that stuff, and as long as I can make it do what I need, it is ok!

--
jacob navia
jacob at jacob point remcomp point fr
logiciels/informatique
<http://www.cs.virginia.edu/~lcc-win32>

.