

## Re: self-confidence of compiler

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-05/msg01263.html>

---

- *From:* "tjmadden1128@xxxxxxxxxx" <tjmadden1128@xxxxxxxxxx>
  - *Date:* 9 May 2007 07:27:07 -0700
- 

On May 8, 5:06 pm, Jon Skeet [C# MVP] <s...@xxxxxxxxxx> wrote:

Peter Duniho <NpOeStPe...@xxxxxxxxxxxxxxxxxxxxxx> wrote:

<snip>

I do know that I don't recall my C++ code generating "this variable is unassigned" errors, but that's probably due to the lack of *\*any\** checking as opposed to more correct checking.

I *\*seem\** to remember that gcc had some checking like this, and Microsoft's C compiler does with /Wall on, but it's not *\*hugely\** smart – it still complains at:

```
#include <stdio.h>
```

```
int main(void)
{
  int x;
  int y=1;
  if (y==1)
  {
    x = 5;
  }
  printf ("%d", x);
}
```

Interestingly, the warning here talks about a "potentially uninitialized local variable" – but if you remove the assignment in the if statement, it turns the error into just "uninitialized local variable".

<snip>

gcc, aCC (HP-UX), and Digital Mars all complain in some fashion about uninitialized variables if you set the warning level high enough. I'd

Re: self-confidence of compiler

have to do some tests as to what the exact messages are. I've had issues in the dark past with static locals being used without being initialized and the compiler not catching that.

Tim

.