

## Re: memset doubt

---

*Source:* [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2005-08/msg03127.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2005-08/msg03127.html)

---

- *From:* "Peter Nilsson" <[airia@xxxxxxxxxxxx](mailto:airia@xxxxxxxxxxxx)>
  - *Date:* 30 Aug 2005 22:59:48 -0700
- 

srivatsan\_b wrote:

> Hi,  
>  
> Can somebody explain whether an explicit typecast is mandatory while  
> calling memset function for a structure? like in the following code  
> snapshot.....  
> struct some\_structure x;  
> memset((some\_structure\*)&x,0,sizeof(some\_structure));  
> Will memset(&x,0,sizeof(some\_structure)); cause some issues?

Slightly better is... `memset(&x, 0, sizeof x);`

In C, there is an issue, but it's not the issue you're asking about.

In C, any object pointer can be implicitly converted to a pointer to void [subject to cv-qualifiers.] So the answer to your question is no. C++ is different, but you should ask in a C++ group if you are actually using a C++ compiler.

Where you may have problems is that setting all the bits of the structure contents to zero, may not do what you want. If your structure contains pointers for instance, then setting the bits to 0 need not set them to null pointers.

If you want to zero initialise a structure (or indeed any other object type), then you simply need to do...

```
struct some_structure x = { 0 };
```

No memset is required.

—  
Peter

.

---

- *References:*

Re: memset doubt

◆ **memset doubt**

◇ *From:* srivatsan\_b

- Prev by Date: **Re: confused abt file operations**
- Next by Date: **Re: memset doubt**
- Previous by thread: **memset doubt**
- Next by thread: **Re: memset doubt**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**