

## Re: A little help please

**Source:** <http://coding.derkeiler.com/Archive/C/ CPP/alt.comp.lang.learn.c-cpp/2004-12/0968.html>

---

**From:** Rich (*Someone\_at\_somewhere.com*)

**Date:** 12/28/04

Date: Tue, 28 Dec 2004 13:02:58 +0000 (UTC)

> *You don't need the cast.*

But according to the question if I read it correctly I do???

The question is exactly as written to newsgroup.

> *Don't do that! Why would you want something like that when there are plenty  
> of alternatives?*

:) It is the question I have to answer

>

> *You cannot do that since arr is of type void\*. You need to cast it to  
> another type. Since func1() has no way of knowing the actual type, this  
> function is invalid.*

Well I guess it would be if I didn't know I was passing an array of ints  
disguised as a void\* :)

>

>

>> *arr += nBytes // increment by nBytes to get next element*

>

>

> *That shows us you don't know how pointers work and that means you shouldn't  
> use them.*

??? if arr points to first element of int array then I thought by  
incrementing the pointer by nBytes I was pointing to next array element.?

> *That's illegal.*

>

> *char ch = ' ';*

>

> *or*

>

> *char ch = 0;*

>

> *depending on what you want.*

```
) this was a typo on my part it was supposed to be char ch = ' ';  
>  
>  
>>int* i = &arrX[0]; // set i as pointer to first element of array  
>>void * vp = static_cast<void*>(i); // convert i to a void*  
>  
>  
> You don't need that  
>  
> void *vp=i;
```

right now I am confused, the question states that I should cast the int array to a void\* using a static cast.

```
>  
>  
>> // assign to vp  
>>func1( vp, sizeof( arrX ), 8 );  
>>for( int i = 0; i < 10; i++ )  
>  
>  
> Prefer ++i over i++.
```

if I did ++i would this not make the start of loop start at 1 ?? since it increments before evaluating?

> That's because you cannot operate on void pointers, you need to cast them.

This is what I wondered about needing to cast back to int array in function?

> Your code is bound to be invalid. You'll have to assume a type (such as  
> char, as does memset()), ask the user for one or forget that function.  
> Templates are meant for this kind of problem.

We have not covered such things yet, we have played with vectors but not templates yet.

```
>  
>  
>>My third question, is am I correct in thinking that the only time to use a  
>>void* is when the variable being assigned could be of any type?  
>  
>  
> I don't understand that.  
>
```

I meant the only reason I could see for using a void\* is for when you don't know the type to be assigned to a pointer so by declaring the pointer void\* I can assign any type to it, but can't dereference a void\* but instead have to cast it back to dereference it.

For example

alt.comp.lang.learn.c-c++: Re: A little help please

```
void* vp;  
vp = &i;
```

```
// *vp = 4 // can't do this can't dereference a void*  
// have to do this instead
```

```
i = static_cast<int>(vp); // cast back to int*  
*i += 4; // i += 4 would make pointer point 4 ints further in memory?
```

Thanks anyway