

## Re: sizeof on arrays and pointers

**Source:** <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.language/2005-01/0799.html>

---

**From:** Doug Harrison [MVP] ([dsh\\_at\\_mvps.org](mailto:dsh_at_mvps.org))

**Date:** 01/17/05

Date: Mon, 17 Jan 2005 10:29:10 -0600

BRG wrote:

>The following program produces different results on the five different  
>compilers I am using:

>

>-----

>#include <stdio.h>

>typedef char ctype[100];

>int main()

>{ char x[100], \*y;

> ctype c;

> printf("\n%3ld %3ld", sizeof(x), sizeof(&x));

> printf("\n%3ld %3ld", sizeof(y), sizeof(&y));

> printf("\n%3ld %3ld", sizeof(c), sizeof(&c));

> return 0;

>}

>-----

>

>These results are as follows:

>

>Compiler 1 Compiler 2 Compiler 3 Compiler 4 Compiler 5

>100 100 100 100 100 4 100 2 100 100

>4 4 4 4 4 2 2 4 4

>100 100 100 100 100 4 100 2 100 100

>

>VC++ produces the first of these and is claimed to be wrong by some who

>think that the right answer is number three.

>

>Is this a bug in VC++?

Yes. The type of the expressions &x and &c above is char (\*)[100], or pointer to array of char of size 100. This bug also affects the Whidbey beta, so you might want to go here and report it:

<http://lab.msdn.microsoft.com/vs2005/default.aspx>

It might also be interesting to print typeid(&x).name(), e.g.

```
// a.cpp

#include <stdio.h>
#include <typeinfo>

int main()
{
    char x[100];
    printf("sizeof(%s) = %d\nsizeof(%s) = %d\n",
        typeid(x).name(),
        (int) sizeof(x),
        typeid(&x).name(),
        (int) sizeof(&x));
    return 0;
}
```

Here is a copy of my Whidbey beta compile session and output for this program:

```
C>cl -W4 a.cpp
Microsoft (R) 32-bit C/C++ Optimizing Compiler Version 14.00.41202 for 80x86
Copyright (C) Microsoft Corporation. All rights reserved.
```

```
a.cpp
Microsoft (R) Incremental Linker Version 8.00.41202
Copyright (C) Microsoft Corporation. All rights reserved.
```

```
/out:a.exe
a.obj
```

```
C>a
sizeof(char [100]) = 100
sizeof(char (*)(100)) = 100
```

```
--
Doug Harrison
Microsoft MVP - Visual C++
```