
Subject: How to fix this strong behaviore of MSC compiler & temporary value ?

Posted by [omari](#) on Fri, 24 Apr 2015 10:56:30 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello:

the code below does not work as expected when compiled with MSC compiler, tested with MSC10/WinXP and MSC12/Win7:

```
#include <Core/Core.h>

using namespace Upp;

class A
{
public:
    int x;
    A():x(0)    {}

    A( A rval_ a)  { LOG("from A");x = a.x;}
    A(String const &s) { LOG("from String");}
    operator String() { LOG("to String"); return ""; }
};

A GetA()
{
    A a;
    a.x = 6;
    return a;
}

CONSOLE_APP_MAIN
{
    A a = GetA();
    DUMP(a.x);
}
```

the output of this test case is :

```
from A
to String
from String
```

a.x = 1310556

Subject: Re: How to fix this strong behaviore of MSC compiler & temporary value ?
Posted by [deep](#) on Sat, 25 Apr 2015 08:07:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Omari

Try this,
Declare first. Then use.
A a ;

```
CONSOLE_APP_MAIN
{
  A a ; // this is changed
  a = GetA();
  DUMP(a.x);
}
```

A a = GetA();
Declaration and assignment on same line does not work particularly function return value.
I don't know why it is this way.

I experienced this behavior with some other functions.

Subject: Re: How to fix this strong behaviore of MSC compiler & temporary value ?
Posted by [omari](#) on Mon, 26 Oct 2015 11:51:59 GMT

[View Forum Message](#) <> [Reply to Message](#)

I think the best solution is to never use "const &", but use rval_

```
class A
{
public:
  int x;
  A():x(0)      {}

  A( A rval_ a) { LOG("from A");x = a.x;}
  A(String rval_ s) { LOG("from String");}
```

```
operator String() { LOG("to String"); return ""; }  
};
```

work as expected.
