
Subject: Re: Order of member initialization
Posted by [peterh](#) on Tue, 20 Sep 2022 13:19:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi I am just a beginner learning (I have practice in C and Object Pascal) and find your example interesting, so I tried it.

It shows zero for A and for B here with CLANGx64 and MSVC22x64. So it seems to work here. (Windows10 x64, Upp build 16332)

So far I have read, the purpose of the initialization list is to initialize variables before any code is run and before the vtables are built.

This is why sometimes lists must be used, so the explanation in my book.

So I believe your example should work as intended.

I modified your example from reference to pointer and it runs equally well.

A pointer in theory can be initialized to an address where the target object does not yet exist (not recommended, but possible)

Now, a reference is essentially a constant self-dereferencing pointer that cannot be modified and must be initialized.

Here my pointer version:

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

```
using namespace Upp;
```

```
class A{
public:
    Array<int> *array;
    A(Array<int> &array_) : array(&array_){
        Cout() << "Array Initial item count in A = " << array->GetCount() << "\n";
    }
};
```

```
class B : public A{
public:
    Array<int> b_array;
    B() : A(b_array){
        Cout() << "Array Initial item count in B = " << b_array.GetCount() << "\n";
    }
};
```

```
CONSOLE_APP_MAIN{
    B b;
    Cout() << "Fertig\n";
    Sleep(1000000);
}
```