Subject: Re: Order of member initialization
Posted by Tom1 on Tue, 20 Sep 2022 15:05:25 GMT
View Forum Message <> Reply to Message

peterh wrote on Tue, 20 September 2022 17:47I must correct myself, it doesnt work.
This code:

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

using namespace Upp;

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

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

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

produces this output:

```
construct A
Array Initial item count in A = 107746496
construct B
Array Initial item count in B = 0
Fertig
```

So constructor A is executed, before B was constructed.

It should however work, if A doesnt use or touch the referenced array before it was constructed.

Hi Peter,

Yes, it does work when the constructor of A does not use or pass the array on to be used by others. However, this may get difficult to control when projects are large.

Best regards,

Tom