

MAUS - Bug #949

complex(double real) should be marked as explicit

21 March 2012 09:36 - Rogers, Chris

Status:	Rejected	Start date:	21 March 2012
Priority:	Normal	Due date:	
Assignee:	Lane, Peter	% Done:	0%
Category:	common_cpp	Estimated time:	0.00 hour
Target version:	Future MAUS release		
Workflow:	New Issue		

Description

Related to 948 - now compiler is trying to call

```
sqrt(Complex z)
```

even when I do

```
double x(sqrt(16.));
```

because C++ has this weird, broken behaviour where it automatically calls constructors to do type conversions in function headers. Use explicit keyword to stop this please (it's in the style guide).

History

#1 - 21 March 2012 09:43 - Rogers, Chris

Humm, not sure what its doing - but to call sqrt() I needed to comment out both:

```
MAUS::Complex MAUS::sqrt(const MAUS::Complex& x)
MAUS::Complex MAUS::sqrt(const double x)
```

#2 - 21 March 2012 15:37 - Lane, Peter

IIRC, because there are matches for sqrt() in the namespace you are in, the compiler stops looking for alternatives. If the <cmath> functions were indeed in the STL library then they would be defined within the std namespace and this wouldn't be a problem. But these functions are in fact throwbacks to C++'s C origins, so they are in the default namespace. You have to explicitly use "::sqrt(x)".

I am fine with removing the "const double x" function (admittedly this was probably a bad idea in the first place), but basically all of the math functions operating on MAUS::Complex arguments will conflict with <cmath> functions. I could rename them all, but I find this distasteful given that it defeats the whole purpose of using namespaces.

Another option might be to make MAUS::Complex a class instead of a typedef of gsl_complex, but that would complicate it's usage elsewhere. I'll have to think about this.

#3 - 21 March 2012 15:40 - Lane, Peter

I meant to say that I could make MAUS::Complex a class and make these functions members of that class. It kind of goes against the convention in C++ of using free functions if you can do the operation with the public class interface, but I'm not going to loose sleep over that.

#4 - 22 March 2012 06:43 - Rogers, Chris

I didn't realised that you could force back to the global namespace like that. I tend to like to group things into a subnamespace (or if they need some global data to work, maybe a class with everything declared static). I didn't realise the workaround though, so maybe it's a non-issue.

#5 - 28 March 2012 07:12 - Rogers, Chris

- Status changed from Open to Rejected

Humm, I'll reject these and just note that folks have to use ::sqrt rather than sqrt...