CoCoALib - Design #619

Modulus (for CRTMill) ambiguous

09 Sep 2014 15:22 - John Abbott

Status:	Closed	**Start date:**	09 Sep 2014
Priority:	Normal	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	Renaming	**Estimated time:**	0.00 hour
Target version:	CoCoALib-0.99536 June 2015	**Spent time:**	2.70 hours

Description

The type CRTMill has an associated function modulus(const CRTMill&); however calling the function the obvious way (*e.g.* modulus(crt)) creates an ambiguity according to the compiler.

This is very irritating. It can be avoided by calling explicitly CoCoA::modulus(crt), but this is not very readable.

Find a better solution.

Related issues:

Related to CoCoALib - Design #620: Redesign CRTMill	**In Progress**	**11 Sep 2014**
Related to CoCoALib - Bug #631: Ambiguous: rank for matrix (in ex-matrix1.C)	**Closed**	**26 Sep 2014**
Related to CoCoALib - Feature #639: Shadow CoCoA namespace to help guarantee ...	**New**	**28 Oct 2014**
Related to CoCoALib - Design #642: Move code in test file into namespace CoCoA	**Closed**	**29 Oct 2014**

History

#1 - 09 Sep 2014 15:25 - John Abbott

I do not understand why the compiler thinks that modulus(crt) is ambiguous.

There is an STL template fn modulus, but I thought template fns were "lower priority" than "exact match" calls. Perhaps its a problem with my old compiler?

I suppose we'll be forced to rename modulus. Any suggestions for a new name?

20141014 correction: in the STL there is a templated class called modulus, so modulus(blah) could also be a ctor call.

#2 - 11 Sep 2014 12:37 - John Abbott

An obvious "solution" is to put using CoCoA::modulus somewhere inside the function which calls it. Just about acceptable, but still a nuisance.

#3 - 14 Oct 2014 18:18 - John Abbott

- Status changed from New to In Progress

- % Done changed from 0 to 10

I think the C++ rules will compel us to change name -- asking the user to write CoCoA::modulus every time is not acceptable. I could not find any useful hints on the internet.

What new name could we use? CombinedModulus (rather long), CurrModulus or CurrentModulus (or CurrMod?)

I prefer not to use a member fn as this is "incompatible" with our philosophy of using them only for modifying operations.

#4 - 14 Oct 2014 18:22 - Anna Maria Bigatti

John Abbott wrote:

> I think the C++ rules will compel us to change name -- asking the user to write CoCoA::modulus every time is not acceptable. I could not find any useful hints on the internet.
>
> What new name could we use? CombinedModulus (rather long), CurrModulus or CurrentModulus (or CurrMod?)

CombinedModulus is long but very clear (better than modulus).
CRTModulus?

#5 - 14 Oct 2014 18:26 - John Abbott

Assuming my explanation in the *correction* in comment 1 is correct, then the STL people could break our code at any time by introducing a new template class whose name coincides with the name of any function we have defined :-(

#6 - 14 Oct 2014 21:34 - John Abbott

(see [#631](#) comment 5) I think the problem goes away if the code is put inside namespace CoCoA.

#7 - 28 Oct 2014 15:44 - John Abbott

- Assignee set to John Abbott

- % Done changed from 10 to 20

As Anna pointed out the names **CombinedModulus** and **CombinedResidue** are very clear (though long). Since it seems unlikely that these functions will be used in complicated formulas/expressions, their length is probably not a hindrance.

I'll implement the changes.

#8 - 28 Oct 2014 16:11 - John Abbott

- Status changed from In Progress to Feedback

- % Done changed from 20 to 90

Implemented; all tests pass. Updated doc.

Oddly, no tests called these fns; nor any examples -- rectify?

Checked in.

#9 - 25 Jun 2015 18:12 - John Abbott

- Status changed from Feedback to Closed

- % Done changed from 90 to 100

Added example and test (actually they are the same).
Checked in. Closing!