

CoCoALib - Bug #22

Rename PPMonoidEvZZ?

08 Nov 2011 09:18 - Anna Maria Bigatti

Status:	Closed	Start date:	08 Nov 2011
Priority:	High	Due date:	
Assignee:	John Abbott	% Done:	100%
Category:	Renaming	Estimated time:	10.00 hours
Target version:	CoCoALib-0.9953	Spent time:	6.15 hours
Description			
Should it be called PPMonoidEvBigInt ?			
(20120530)			
Note at the moment the class is called PPMonoidBigEv and the file is called PPMonoidEvZZ -- just so you know!			
Related issues:			
Related to CoCoALib - Support #774: Rename file PPMonoidEvZZ		Closed	15 Sep 2015

History

#1 - 08 Nov 2011 09:42 - Anna Maria Bigatti

- Project changed from CoCoA to CoCoALib

#2 - 08 Nov 2011 09:48 - Anna Maria Bigatti

- Category set to Renaming

#3 - 09 Nov 2011 15:15 - Anna Maria Bigatti

- Target version set to CoCoALib-0.9950

#4 - 09 Nov 2011 15:17 - Anna Maria Bigatti

- Estimated time set to 2.00 h

#5 - 18 Mar 2012 15:19 - John Abbott

Here is an alternative suggestion (affecting the names of all PPMonoids).

The underlying principle is that an object which imposes limitations should have a name which indicates that it is not fully general (thereby presumably encouraging a potential user to check the documentation...)

Currently all PPMonoids (except PPMonoidEvZZ) impose limitations on the maximum degree of a PP. It would make more sense if the degree limited PPMonoidEv had a name something like PPMonoidEv_SmallExp. Then we could also have PPMonoidEv_BigExp.

The length of the class names is not important to a normal CoCoALib user who sees only the pseudo-ctors NewPPMonoidEv. We could merge NewPPMonoidEv and NewPPMonoidEvZZ into a single pseudo-ctor which takes an optional arg saying whether we want "big" or "small" exponents -- strictly following my reasoning above the default ought to be "big" exponents (but here I see a potential clash between ease of use and "correctness").

Inside the .C file we could create two (virtually identical?) classes choosing whatever names we like. It might make sense to put these classes inside a namespace, so we'd actually have PPMonoidEv::SmallExpImpl and PPMonoidEv::BigExpImpl (for instance). If the impls really are virtually identical we ought to consider a template class implementation; at the moment I do not expect the impls to be sufficiently similar (because one packs the exponents into integers, the other will probably not use any such packing).

We could also consider a "very small exponent" case too. This might give some slight space advantage, but probably no significant variation in computation time. Note that currently the integer type used for exponent values is determined by a typedef in the file config.H. Personally I doubt that a "very small exp" version will be useful because new computers seem to have lots of RAM (8Gbytes in a portable is already available); in any case a perceptible gain would arise only if the coeffs are in a small finite field and there are lots of indets.

#6 - 30 Mar 2012 14:34 - John Abbott

- Target version changed from CoCoALib-0.9950 to CoCoALib-0.9951

We don't have time to complete this change before 0.9950.

However we have decided that it is simplest for the user if the pseudo-ctor has a single name and an optional parameter saying whether we want big or small exponents -- a similar change will be made to all PPMonoid pseudo-ctors.

Since the changes to be made are non-trivial and we want to complete 0.9950 as soon as possible the target version has now become 0.9951.

#7 - 30 Mar 2012 14:36 - John Abbott

We have not yet decided about internal impl names.
JAA still likes his suggestion made two posts ago.

#8 - 28 May 2012 16:20 - John Abbott

A "very small exp" version may be slightly faster because more PPs fit in the cache, but I doubt the slight gain would be worth the extra complication in the source.

#9 - 28 May 2012 17:16 - John Abbott

- Target version changed from CoCoALib-0.9951 to CoCoALib-0.9952

#10 - 18 Feb 2013 16:34 - John Abbott

- Status changed from New to In Progress
- Assignee set to John Abbott
- Priority changed from Normal to High
- % Done changed from 0 to 10
- Estimated time changed from 2.00 h to 10.00 h

We have agreed to implement the decision in note 6.

#11 - 14 Mar 2013 14:29 - John Abbott

- Target version changed from CoCoALib-0.9952 to CoCoALib-0.9953

#12 - 15 Mar 2013 12:12 - John Abbott

- Status changed from In Progress to Feedback
- % Done changed from 10 to 90

The class names are not public, so i have made them descriptive (and thus long).
I have changed the interface of the pseudo-ctor NewPPMonoidEv: there is a new optional final arg to say whether the exps will be big or small (default is small).

All tests pass; Anna approves the new interface. Code has been checked in.
Doc has been updated.

#13 - 23 May 2013 16:58 - John Abbott

- *Status changed from Feedback to Closed*

- *% Done changed from 90 to 100*

Anna and I have checked that everything is present and correct.
So we close this issue.