

CoCoALib - Bug #89

MachineInt or long as fn arg type for indices

09 Feb 2012 22:51 - John Abbott

Status:	Closed	Start date:	09 Feb 2012
Priority:	High	Due date:	
Assignee:	John Abbott	% Done:	100%
Category:	Tidying	Estimated time:	8.00 hours
Target version:	CoCoALib-0.9953	Spent time:	2.00 hours
Description In file PPMonoid.H several functions expect a long arg to indicate the index of an indet. Should these arg types be changed to MachineInt? Note that a fn which expects a vector of such indices will have an arg type of vector<long> If we decide that long should be changed in MachineInt, then we must make the change. I'm sure a similar problem exists in several other header files; we must check these too (but perhaps list them as separate tasks?)			
Related issues:			
Related to CoCoALib - Feature #124: change long args in matrices into Machine...		Rejected	04 Apr 2012
Related to CoCoALib - Design #925: MachineInt or long for args which are indi...		In Progress	20 Sep 2016

History

#1 - 21 Feb 2013 17:58 - John Abbott

- Category set to Tidying
- Assignee set to John Abbott
- Priority changed from Normal to High
- Target version set to CoCoALib-0.9953

Empty post -- just to "wake up" the issue.

#2 - 25 May 2013 10:46 - John Abbott

- Status changed from New to In Progress
- % Done changed from 0 to 50

JAA thinks it is reasonable to handle indices differently from "arithmetic integers".

JAA suggests "keeping it simple" (but reserving the right to revisit the question if unforeseen problems arise).

Consequently, JAA proposes using **long** for all indices -- this allows negative indices (possibly sensible in some contexts), and anyway I prefer to avoid unsigned types as I've been burned by them too many times!

While MachineInt is a safer option, it is more complicated/cumbersome and does incur a run-time penalty. I think it also makes the source code less readable.

Any contrary opinions? React quickly as I'd like to close this issue ASAP!

TO DO In any case, the CoCoALib source code must be checked for adherence to the chosen type for indices.

#3 - 25 May 2013 11:51 - Anna Maria Bigatti

While MachineInt is a safer option, it is more complicated/cumbersome and does incur a run-time penalty. I think it also makes the source code less readable.

I agree

#4 - 30 May 2013 16:31 - John Abbott

- *Status changed from In Progress to Feedback*
- *% Done changed from 50 to 90*

I have checked/modified all the following:

- matrix indices
- PPMonoid indet indices
- PolyRing indet indices
- operator[] arg
- degree indices
- function coeff for DenseUPolyRing

The following were left as MachineInt:

- indexes in **symbol**
- exponents in powering fns (and fns for extracting roots)

#5 - 30 May 2013 16:40 - John Abbott

The design principle is that indices should be of type long; note that this **includes** *size specifications* given to constructors of indexable objects (such as matrices) or to resizing fns.

#6 - 31 May 2013 16:53 - John Abbott

- *Status changed from Feedback to Closed*
- *% Done changed from 90 to 100*

Since the actual changes I had to make were minimal (i.e. we had anyway used long for almost all instances of indices). I'm regarding this as already sufficiently tested, so I'm closing.

#7 - 20 Sep 2016 18:59 - John Abbott

- *Related to Design #925: MachineInt or long for args which are indices (yet again) added*