

CoCoALib - Feature #1235

FreeModule: get canonical basis

11 Dec 2018 15:40 - John Abbott

Status:	Closed	Start date:	11 Dec 2018
Priority:	Normal	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	New Function	Estimated time:	1.75 hour
Target version:	CoCoALib-0.99800	Spent time:	1.80 hour
Description			
It would be helpful to have a function which produces a standard basis for a FreeModule (in the expected order).			

History

#1 - 11 Dec 2018 15:43 - John Abbott

The function gens for a submodule of a freemodule produces a set of generators (but with no special guarantees). There should also be a function which produces a list of module elements being the standard basis for a freemodule.

It should be fairly trivial to implement.

#2 - 12 Dec 2018 07:19 - Anna Maria Bigatti

Two ideas:
1 - basis(FM);
2 - gens(FM); (with a comment in the manual saying that, for a FreeModule, returns the standard basis)

#3 - 23 Sep 2019 12:45 - John Abbott

- Target version changed from CoCoALib-0.99650 November 2019 to CoCoALib-0.99700

#4 - 08 Jan 2020 23:02 - John Abbott

- Target version changed from CoCoALib-0.99700 to CoCoALib-0.99800

#5 - 03 Nov 2021 19:05 - John Abbott

If we want a special fn for getting the standard/canonical basis for a free module then it could be called **CanonicalBasis**; this fn would throw if the arg is not a free module.

Probably gens applied to a free module should return the canonical basis, but I'm unsure whether we should state this in the documentation.

An advantage of a new special fn is that whoever is reading the code knows that the arg should/must be a free module; whereas if the code calls gens then whoever is reading the code cannot "clearly assume" that the arg is a free module (at least, not based on this fn call).

#6 - 05 Nov 2021 15:42 - John Abbott

- Subject changed from FreeModule: get a standard basis to FreeModule: get canonical basis

- Status changed from New to In Progress

- % Done changed from 0 to 20

Agreed: the proposal in comment 5 is accepted.

To clarify the order of the elements in the canonical basis should be as expected: namely, $(1,0,0,0,\dots)$, $(0,1,0,0,\dots)$, etc

Arg type should be FreeModule (const ref, probably)

#7 - 05 Nov 2021 15:44 - Anna Maria Bigatti

- Assignee set to Anna Maria Bigatti

#8 - 05 Feb 2022 19:41 - John Abbott

- Status changed from In Progress to Resolved

- % Done changed from 20 to 80

I have made a trivial impl (& written short doc).
Must also make available from CoCoA-5 (with doc).

#9 - 14 Feb 2022 18:25 - John Abbott

- Status changed from Resolved to Closed

- % Done changed from 80 to 100

- Estimated time set to 1.75 h

All done. Closing.