# CoCoA-5 - Bug #1640

# MinSubsetOfGens does not find min subset

08 Dec 2021 00:08 - John Abbott

Status:	Closed	Start date:	08 Dec 2021
Priority:	Normal	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	bug	Estimated time:	1.66 hour
Target version:	CoCoA-5.4.2	Spent time:	1.63 hour
Description			

# Description

Not sure what MinSubsetOfGens does, but it does not find a min card subset of the gens which generate the same ideal

# **Related issues:**

Related to CoCoA-5 - Slug #1638: MinSubsetOfGens sometimes very slow

06 Dec 2021

New

#### History

#### #1 - 08 Dec 2021 00:09 - John Abbott

I had wanted to show MinSubsetOfGens to my students, but maybe it is better not to do so.

The code is weird; I am not sure what it does.

# #2 - 08 Dec 2021 00:11 - John Abbott

I have now re-read the manual, and understood what it was trying to say. Maybe re-write the manual entry to make it clearer?

# #3 - 09 Dec 2021 10:18 - Anna Maria Bigatti

- Description updated

# #4 - 09 Dec 2021 10:27 - Anna Maria Bigatti

John Abbott wrote:

I have now re-read the manual, and understood what it was trying to say. Maybe re-write the manual entry to make it clearer?

### Better like this?

NOTE: there no guarantee that  ${\rm <tt>S</tt>}$  is the minimal subset with smallest cardinality.

# #5 - 10 Dec 2021 15:16 - John Abbott

- Related to Slug #1638: MinSubsetOfGens sometimes very slow added

### #6 - 10 Dec 2021 15:53 - Anna Maria Bigatti

During skype meeting, we implemented an improvement (discarding longest polynomial first). Now the given example is much faster (and returns the original 2 generators).

### #7 - 10 Dec 2021 15:54 - Anna Maria Bigatti

- Status changed from New to Feedback
- Assignee set to Anna Maria Bigatti
- % Done changed from 0 to 80

# #8 - 20 Jan 2022 20:32 - John Abbott

- % Done changed from 80 to 90

Unfortunately I have lost the specific example (but I suppose a new one could easily be created). I did notice the following "phenomenon": let RGB be a reduced GB ordered so that LTs are increasing (this is what CoCoA does anyway?)

- it was slow checking if first 2, first 3, first 4 etc gens produce the same ideal;
- it was quite fast to check if all-but-last, all-but-last-2, etc produce the same ideal.

We should test this before closing.

### #9 - 21 Jan 2022 10:10 - John Abbott

- Status changed from Feedback to Closed
- % Done changed from 90 to 100
- Estimated time set to 1.66 h

We had effectively already taken into account the comment 5 above.

Anna has just changed the code slightly to order by LT (previously ordered based on NumTerms).

Closing.