

## CoCoALib - Bug #1779

### Radical error with lex (again)

05 Feb 2024 17:57 - Anna Maria Bigatti

<b>Status:</b>	Closed	<b>Start date:</b>	05 Feb 2024
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Anna Maria Bigatti	<b>% Done:</b>	100%
<b>Category:</b>	Various	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	CoCoALib-0.99850	<b>Spent time:</b>	3.25 hours
<b>Description</b>			
This problem resurfaced now that radical is in C++.			
<pre>/**/ use QQ[x,y],lex; /**/ I := ideal(x*y +x, x^2 +x); /**/ radI := radical(I); --&gt; triggered error --&gt; ERROR: Ring is not standard graded --&gt; [CoCoALib] dim</pre>			
I'm very grateful to test/exbugs and to its reference to redmine with the solution ;-)			
<b>Related issues:</b>			
Related to CoCoA-5 - Bug #1577: radical: lex order, ERROR: Ring is not stand...		<b>Closed</b>	<b>15 Feb 2021</b>
Related to CoCoALib - Slug #1646: radical: could be more clever		<b>Closed</b>	<b>17 Jan 2022</b>
Related to CoCoALib - Feature #1780: radical for ideals in SparsePolyRing: c...		<b>Closed</b>	<b>06 Feb 2024</b>
Related to CoCoALib - Support #1782: Make CoCoALib test file for radical		<b>New</b>	<b>23 Feb 2024</b>

### History

#### #1 - 05 Feb 2024 17:58 - Anna Maria Bigatti

- Related to Bug #1577: radical: lex order, ERROR: Ring is not standard graded added

#### #2 - 05 Feb 2024 17:58 - Anna Maria Bigatti

- Subject changed from Radical error with lex to Radical error with lex (again)

#### #3 - 05 Feb 2024 18:32 - Anna Maria Bigatti

- % Done changed from 0 to 30

first bug fixed.

New bug:

```
/**/ use ZZ/(5)[x,y,z];
/**/ SetVerbosityLevel(99);
/**/ radI := radical(ideal(x^2+z,y+z)); --> gave error in factorizer
--> ERROR: Poly must be non-constant
--> [CoCoALib] UnivariateIndetIndex
```

in the code there is

```
factorization<RingElem> Q = factor(one(RingOf(I))); // sichtbar machen
```

which is a trick. Investigating why and how to avoid it

**#4 - 05 Feb 2024 19:00 - Anna Maria Bigatti**

- Related to Slug #1646: radical: could be more clever added

**#5 - 05 Feb 2024 21:29 - John Abbott**

- Status changed from New to In Progress

It would be easy to allow the factorize to return a result if the input is a non-zero constant. I suppose I made it trigger an error because it is fairly pointless factorizing a constant. Should I change the code to allow factorizing a constant?

**UPDATE** after skype discussion, we have decided to leave **factorizing a deg 0 poly as an error**

**#6 - 06 Feb 2024 09:02 - Anna Maria Bigatti**

- Related to Feature #1780: radical for ideals in SparsePolyRing: code in C++ added

**#7 - 06 Feb 2024 10:12 - Anna Maria Bigatti**

John Abbott wrote:

It would be easy to allow the factorize to return a result if the input is a non-zero constant. I suppose I made it trigger an error because it is fairly pointless factorizing a constant. Should I change the code to allow factorizing a constant?

Right now I'm chasing another problem. But I'll try to understand why there was that line, and then see whether that initialization made sense in that context.

**#8 - 06 Feb 2024 16:47 - Anna Maria Bigatti**

- Status changed from In Progress to Feedback

- % Done changed from 30 to 80

Anna Maria Bigatti wrote:

John Abbott wrote:

It would be easy to allow the factorize to return a result if the input is a non-zero constant. I suppose I made it trigger an error because it is fairly pointless factorizing a constant. Should I change the code to allow factorizing a constant?

I don't know why there was that initialization. Now I call directly

```
factorization<RingElem> Q = SqFreeFactor(G); // redmine #1779.#3
```

And I haven't met any problem with the tests I have.

It seems all fine.

I have another set of tests (CoCoALib tests) which need some polishing. Then I'll close this issue.

**#9 - 23 Feb 2024 11:39 - Anna Maria Bigatti**

- *Related to Support #1782: Make CoCoALib test file for radical added*

**#10 - 23 Feb 2024 11:41 - Anna Maria Bigatti**

- *% Done changed from 80 to 90*

I made a new issue for implementing and running the tests.

Closing this one ... hoping non errors will come from the tests.

**#11 - 23 Feb 2024 12:13 - John Abbott**

- *% Done changed from 90 to 80*

**#12 - 23 Feb 2024 12:28 - Anna Maria Bigatti**

- *Status changed from Feedback to Closed*

- *% Done changed from 80 to 100*