CoCoA-5 - Bug #1062

IsRadical bug?

09 May 2017 09:43 - John Abbott

Status:	Closed	**Start date:**	09 May 2017
Priority:	Normal	**Due date:**	
Assignee:	Anna Maria Bigatti	**% Done:**	100%
Category:	Incomplete function	**Estimated time:**	0.00 hour
Target version:	CoCoA-5.2.2	**Spent time:**	4.45 hours

Description

I have an example where IsRadical produces an error (when applying a RingHom)
0-dimensional bug: related with MinPolyModular, moved to #1064

And this triggers SEGV (non-0-dim bug)

```
use R ::= QQ[X0,X1,X2,X3,X4,X5,X6];
I := ideal(X3^3 +X0*X3*X4 +X4^3,
  X0*X1*X3 +X3*X5^2,
  -X0^6*X1*X3 +X3*X6^7,
  X0 -X1 -X2 -X3 -X5 -X6,
  -X2 +X5 +X6,
  -X0 -X5 -X6,
  X1 -X2 +X3 -X6,
  X1 -X2 +X3 -X5);
radical(I);
```

Related issues:

Related to CoCoALib - Design #1063: Catching an (expected) error	**Closed**	**09 May 2017**
Related to CoCoALib - Bug #1064: Bug in MinPolyModular (ugly prime)	**Closed**	**10 May 2017**

History

#1 - 09 May 2017 09:44 - John Abbott

This example has simpler coeffs:

```
ideal(X3^3 +X0*X3*X4 +X4^3,  8*X0*X1*X3 +4*X0*X2*X3 +6*X0*X4^2 -6*X1*X4^2 +3*X0*X3*X5 +X3*X5^2,  5*X0^6*X1*X3 -8*X0^6*X2*X3 +6*X0^6*X4^2 -36*X0^5*X2*X4^2 +90*X0^4*X2^2*X4^2 -120*X0^3*X2^3*X4^2 +90*X0^2*X2^4*X4^2 -36*X0*X2^5*X4^2 +6*X2^6*X4^2 +4*X0^6*X3*X6 +X3*X6^7,  4*X0 -2*X1 +X2 -2*X3 +3*X4 +8*X5 -8*X6,  -X0 +7*X1 -8*X2 +8*X3 -7*X5 -X6,  -X0 -5*X1 -7*X2 -X3 -6*X4 -9*X5 -9*X6,  4*X0 -3*X1 +7*X2 -9*X3 -X4 -3*X5 -2*X6,  4*X0 -6*X1 -X2 -2*X3 +5*X4 -4*X5 +9*X6)
```

#2 - 09 May 2017 09:47 - John Abbott

Aaargh!! Now I have a SEGV :-(

```
use R ::= QQ[X0,X1,X2,X3,X4,X5,X6];

define RndCoeff()
  return random(-1,1);
enddefine; -- RndCoeff
```

```
StartTime := CpuTime();
for i := 1 to 10000 do

J := ideal(
  X0 * X3 * X4 + X3^3 + X4^3,
  X3 * X5^2 + RndCoeff() * X0 * X3 * X5 + RndCoeff() * X0 * X1 * X3 + RndCoeff() * X0 * X2 * X3 + RndCoeff() *
 (X1 - X0) * X4^2,
  X3 * X6^7 + RndCoeff() * X0^6 * X3 * X6 + RndCoeff() * X0^6 * X1 * X3 + RndCoeff() * X0^6 * X2 * X3 +RndCoef
f() * (X2 - X0)^6 * X4^2,
           sum([RndCoeff()*x | x in indets(R)]),
           sum([RndCoeff()*x | x in indets(R)]),
           sum([RndCoeff()*x | x in indets(R)]),
           sum([RndCoeff()*x | x in indets(R)]),
           sum([RndCoeff()*x | x in indets(R)])
);

  if IsRadical(J) then println "RADICAL: ", J; endif;
endfor;
EndTime := CpuTime();
println "Loop time: ", FloatStr(EndTime-StartTime);
```

#3 - 09 May 2017 09:49 - John Abbott

This example triggers SEGV:

```
ideal(X3^3 +X0*X3*X4 +X4^3,  X0*X1*X3 +X3*X5^2,  -X0^6*X1*X3 +X3*X6^7,  X0 -X1 -X2 -X3 -X5 -X6,  -X2 +X5 +X6,
 -X0 -X5 -X6,  X1 -X2 +X3 -X6,  X1 -X2 +X3 -X5);
```

#4 - 09 May 2017 11:54 - Anna Maria Bigatti

I'm not that surprised. Radical for non zero-dimensional ideals is not that robust.
I'm investigating

#5 - 09 May 2017 11:58 - Anna Maria Bigatti

- % Done changed from 0 to 10

Smaller example

```
use R ::= QQ[X1,X3,X4];
J := ideal(X1 +X3,  X3^3 +X4^3);
```

```
radical(J);
```

#6 - 09 May 2017 15:13 - Anna Maria Bigatti

- Assignee set to Anna Maria Bigatti

- % Done changed from 10 to 30

The two bugs (0-dim and non-0-dim) are quite distinct.

Debugging the zero-dimensional example in the description of this issue was relatively easy: this is the first time we actually stumble on an ugly prime!!

So I just had to put the apply of the partial homomorphism inside a try/catch.
Such an easy thing to say and such a tricky thing to do.

I would like to have a *standard way* to catch some expected error (I'll put this in a related issue).

#7 - 09 May 2017 15:44 - Anna Maria Bigatti

- Related to Design #1063: Catching an (expected) error added

#8 - 10 May 2017 13:16 - John Abbott

- Status changed from New to In Progress

When this is fixed remember to add some new tests to exbugs.cocoa5!

#9 - 10 May 2017 15:37 - Anna Maria Bigatti

- Related to Bug #1064: Bug in MinPolyModular (ugly prime) added

#10 - 10 May 2017 15:43 - Anna Maria Bigatti

- Description updated

#11 - 18 May 2017 07:40 - Anna Maria Bigatti

- % Done changed from 30 to 40

Bug (radical >0 dim) trapped by Elisa Palezzato

```
/**/ saturate(ideal(x -y), ideal(one(CurrentRing)));
```

```
Process cocoa5 segmentation fault: 11
```

#12 - 18 May 2017 07:58 - Anna Maria Bigatti

Found it in TmpGOperations.C:
Saturation by ideal(f) calls factor(f) and wrongly assumes that the factor list contains at list one element.
Fixed it by intercepting IsOne(f)

UPDATE fixed better by using IsInvertible o/w saturate(ideal(x-y, ideal(2*one(P))) gives SEGV

#13 - 18 May 2017 08:01 - Anna Maria Bigatti

- Description updated

#14 - 18 May 2017 08:16 - Anna Maria Bigatti

- Status changed from In Progress to Feedback

- % Done changed from 40 to 90

added tests. CVS-ed

#15 - 22 May 2017 22:41 - John Abbott

Does your test of saturate in anna.cocoa5 compute a saturation with twin-float coeffs??
I think you might want to insert a use command a line 143. Right?

#16 - 23 May 2017 08:36 - Anna Maria Bigatti

John Abbott wrote:

> Does your test of saturate in anna.cocoa5 compute a saturation with twin-float coeffs??

It was not intentional, but why not?

#17 - 10 Nov 2017 12:18 - John Abbott

- Status changed from Feedback to Closed

- % Done changed from 90 to 100