CoCoALib - Bug #1449

Bivariate factor bug: sometimes a factor is reducible

20 Apr 2020 13:54 - John Abbott

Status:	Closed	**Start date:**	20 Apr 2020
Priority:	Urgent	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	Maths Bugs	**Estimated time:**	5.99 hours
Target version:	CoCoALib-0.99710	**Spent time:**	5.85 hours

Description

The following code prints out some faulty (*i.e.* not irred) factorizations:

```
use QQ[x,y];

p1 := (x+y)*(500*y^8 +400*y^7*x +1283*y^6*x^2 +2620*y^5*x^3 +2474*y^4*x^4 +2620*y^3*x^5 +1283*y^2*x^6 +400*y*x^7 +500*x^8 -10800*y^6 -18144*y^5*x -29808*y^4*x^2 -44928*y^3*x^3 -29808*y^2*x^4 -18144*y*x^5 -10800*x^6 +85968*y^4 +177984*y^3*x +184032*y^2*x^2 +177984*y*x^3 +85968*x^4 -297216*y^2 -501120*y*x -297216*x^2 +373248);

for j := 1 to 100 do
facs := factor(p1);
  if len(facs.multiplicities) <> 3 then println [j,facs];  endif;
endfor;
```

Reported by Jasper Nalbach (from Aachen)

Related issues:

Related to CoCoALib - Support #1338: Release CoCoALib-0.99700	**Closed**	**15 Oct 2019**

History

#1 - 20 Apr 2020 13:56 - John Abbott

I have simplified slightly the problem (from what Nalbach reported by email).

About 7% of calls gave a wrong result (after running a loop with 1000 iters).
Almost certainly an "unlucky random number" (but that should never happen...)

#2 - 20 Apr 2020 15:32 - John Abbott

- Status changed from New to In Progress

- Assignee set to John Abbott

- % Done changed from 0 to 10

Here is a simpler failing example:

```
p1 := (x+y)*(2*x^2 +x*y +2*y^2 -12)*(120*x^6 +250*y^6  +168*x*y  -104);
```

I'll try making it simpler still. If I set coeffs of x^6 and y^6 both to 1 then it seems to work fine...

#3 - 21 Apr 2020 10:18 - John Abbott

Here is a simpler example:

```
p1 := (x+y)*(x^2 +y^2)*(2*x^6 +y^6 +2);
```

I have tried reducing the exponent 6 but without success (*i.e.* the factorization comes out right).

This is probably simple enough that I should now start the hard phase... 8-{

#4 - 22 Apr 2020 21:57 - John Abbott

- % Done changed from 10 to 20

The problem is in the **univariate** factorizer (ouch!)

```
use QQ[x,y];
jj := (x+41)*(x^2+41^2)*(2*x^6+41^6+2); // subst(p1,y,41);
factor(jj); --> only 2 factors!!

p1 := (x+y)*(x^2 +y^2)*(2*x^6 +y^6+2);
subst(p1,y,37); // or -37, also goes wrong
subst(p1,y,41); // or -41, also goes wrong
```

#5 - 23 Apr 2020 08:54 - Anna Maria Bigatti

John Abbott wrote:

> The problem is in the **univariate** factorizer (ouch!)
>
> [...]

I confirm on my Mac: 2 factors :-(

#6 - 24 Apr 2020 15:53 - John Abbott

- % Done changed from 20 to 30

I have found a bug... not yet sure how to fix it. :-/
The crucial line is around DUPZfactor_combine.c:315; the value of max_tuple_size can be incorrectly set to THIS->nfactors/2.
We have modular factors of degs 1,1,1,6, and have just found a linear factor; so remaining modular factors have degs 1,1,6, and for some reason the code thinks the search does not need to consider 2-tuples... logic error!

Here is a simpler univariate problem poly:

```
f := (x+41)*(x^2+41^2)*(2*x^6+1608941);
```

#7 - 24 Apr 2020 16:36 - John Abbott

- % Done changed from 30 to 40

Here is an even simpler example:

```
f := (x+11)*(x^2+11^2)*(2*x^4+235885);
```

```
f := (x+11)*(x^2+1)*(2*x^4+235771); -- 2*x^7 +22*x^6 +2*x^5 +22*x^4 +235771*x^3 +2593481*x^2 +235771*x +259348
1
```

And some even simpler examples:

```
flist := [(x+11)*(x^2+1)*(x^4+n) | n in [21458,21469,21470,21484,21496,21497]];
```

UPDATE I did a "quick" search for bad polys of the form (x+m)*(x^2+1)*(n1*x^3+n2) but found none (for m=11,..,20, n1=1,..,10 and n2=1,..,250000)

#8 - 24 Apr 2020 20:43 - John Abbott

- Status changed from In Progress to Resolved

- % Done changed from 40 to 70

I has disabled the dodgy looking short-cut, and now several tests (incl. the original example) have worked just fine.

I guess that ancient factorizer code needs to be brought up to date... sigh!

#9 - 25 Apr 2020 14:31 - John Abbott

- Status changed from Resolved to Feedback

- % Done changed from 70 to 90

- Estimated time set to 5.55 h

I have done some ad hoc testing; seems OK.
Checked in; incr version number of CoCoALib (CoCoA-5 too).

ANNA: Could you check on you machine? Thanks.

#10 - 26 Apr 2020 19:43 - John Abbott

Added test to exbugs.cocoa5.

#11 - 26 Apr 2020 19:57 - John Abbott

- Estimated time changed from 5.55 h to 5.99 h

Added test to src/tests/test-bug1.C

#12 - 29 Apr 2020 12:43 - John Abbott

- Related to Support #1338: Release CoCoALib-0.99700 added

#13 - 29 Apr 2020 12:43 - John Abbott

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

#14 - 30 Apr 2020 14:19 - Redmine Admin

- Target version changed from CoCoALib-0.99700 to CoCoALib-0.99710

#15 - 20 May 2020 13:26 - John Abbott

- Status changed from Feedback to Closed

- % Done changed from 90 to 100