CoCoALib - Feature #979

SmallestNonDivisor -- new fn

21 Nov 2016 17:51 - John Abbott

Status:	Closed	**Start date:**	21 Nov 2016
Priority:	Normal	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	New Function	**Estimated time:**	0.00 hour
Target version:	CoCoALib-0.99560	**Spent time:**	2.35 hours

Description

In the context of the "universal denominator" for a polynomial ideal, it could be useful to find the first prime which does not divide this number.

A possible name is SmallestNonDivisor.
What exact semantics do we want?

Related issues:

Related to CoCoA-5 - Support #977: "universal denominator" (related with Groe...	**In Progress**	**17 Nov 2016**

History

#1 - 21 Nov 2016 17:51 - John Abbott

- Related to Support #977: "universal denominator" (related with GroebnerFanIdeals) added

#2 - 21 Nov 2016 18:01 - John Abbott

After a brief discussion on email, I think it is best to write on redmine.
Robbiano proposes a function to find the "first/smallest" prime which does not divide the "universal denominator".

Proposed names are: SmallestNonDivisor, FirstNonDivisor, FirstExcellentPrime.
What should the arg(s) be? Obviously an arg must be the number (BigInt) whose prime divisors we must avoid; a possible second arg could be a lower bound for the prime. Result can be a long.

JAA wonders whether it might be a good idea to have an iterator which generates a succession of non-dividing primes. A suitable name could be PrimeNonDivisors. What would the arg(s) be? Just the BigInt whose prime divisors we wish to avoid? Or perhaps also a lower bound for the prime non-divisors? The iterator would produce long values.

NOTE a quick test in C++ suggests that a simplistic loop works tolerably well. It does not appear to be worth the trouble of trying to remove all powers of prime divisors. Anyway, extreme speed is probably not so important.

#3 - 22 Nov 2016 15:31 - John Abbott

- Status changed from New to Resolved

- Assignee set to John Abbott

- % Done changed from 0 to 80

Added **SmallestNonDivisor** to NumTheory; also available in CoCoA-5.
Updated doc for NumTheory.
Still to do: update doc for CoCoA-5, example, test.

#4 - 22 Nov 2016 15:38 - John Abbott

I still like the idea of an iterator, but it would be hard to "export" the notion to CoCoA-5. Maybe that does not matter so much?

#5 - 22 May 2017 13:38 - Redmine Admin

- Target version changed from CoCoALib-1.0 to CoCoALib-0.99560

#6 - 22 May 2017 13:52 - John Abbott

While preparing an example of a simple ideal with a big "universal denominator" I encountered a number with about 330 decimal digits which I wanted to factorize.

It turned out that a better approach was to take all denominators of all coeffs of all polys in all RGBs and then call GCDFreeBasis. Finally I had to use SmoothFactor on the few numbers which were not IsPrime; this allowed me to obtain the full factorization (with some primes > 10^10).

Could there be any interest/utility in trying to produce a factorized form of the universal denominator?

#7 - 06 Nov 2017 15:12 - John Abbott

- Status changed from Resolved to Closed

- % Done changed from 80 to 100

This is apparently complete (also included in CoCoA-5).