CoCoALib - Feature #1154

SmallFpImpl: new ctor arg to say do-not-check-that-arg-is-prime

11 Feb 2018 20:49 - John Abbott

Status:	Closed	**Start date:**	11 Feb 2018
Priority:	Normal	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	New Function	**Estimated time:**	1.99 hour
Target version:	CoCoALib-0.99600	**Spent time:**	1.90 hour

Description

I propose adding a new ctor for SmallFpImpl where the caller can use a flag to guarantee that the arg is prime.

Reason: testing a number for primality is not so cheap (esp. for numbers over 1000000000), so some CRT loops spend more time checking numbers for primality than actually computing the answer! *e.g.* JAA tried DetByCRT on a 4x4 matrix with large integer entries (30000 digits)

Related issues:

Related to CoCoALib - Feature #797: SmallFpImpl: make it faster	**In Progress**	**07 Nov 2015**
Related to CoCoALib - Feature #1155: Create a new "prime source" iterator	**Closed**	**11 Feb 2018**

History

#1 - 11 Feb 2018 20:52 - John Abbott

Several CRT loops look a lot like this:

```
  while (true)
  {
    p = NextPrime(p);
    ModP = SmallFpImpl(p);
    // do computation mod p
  }
```

The point is that both NextPrime and SmallFpImpl check that the number is prime, and this is quite costly (when the number actually is prime).

So maybe there should be a ctor SmallFpImpl(p, NoCheck) which says not to check that arg is prime (woe betide those who lie!)

#2 - 11 Feb 2018 20:52 - John Abbott

- Related to Feature #797: SmallFpImpl: make it faster added

#3 - 11 Feb 2018 21:07 - John Abbott

- Related to Feature #1155: Create a new "prime source" iterator added

#4 - 11 Feb 2018 21:10 - John Abbott

Another posibility is for a "prime source" to produce values of a new type SmallPrime (which is really just a long, but with the guarantee that its arg is prime). Then there could be ctor for SmallFpImpl which accepts a SmallPrime, and knows that it does not need to check primality!

#5 - 25 Jun 2018 14:10 - John Abbott

- Status changed from New to Feedback

- Assignee set to John Abbott

- % Done changed from 0 to 90

I think that SmallPrime solves this matter reasonably well. It does require making 2 ctors (copy-and-paste), but they are fairly short and simple.

Changed to **Feedback**. Will check-in shortly. Maybe I should update the doc?

#6 - 03 Aug 2018 16:14 - John Abbott

- Status changed from Feedback to Closed

- % Done changed from 90 to 100

- Estimated time set to 1.99 h

Aim effectively achieved by the new class SmallPrime
(this is a cleaner and more general solution than one originally proposed).
All working fine for the last 6 months -- so closing.