CoCoA-5 - Bug #158

May AsRAT produce an INT?

14 May 2012 14:45 - John Abbott

Status:	Closed	**Start date:**	14 May 2012
Priority:	Normal	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	CoCoA-5 function: new	**Estimated time:**	2.75 hours
Target version:	CoCoA-5.0.3	**Spent time:**	2.75 hours

Description

What type should the result of **AsRAT(3)** have?

At the moment AsRAT returns an **INT** if its arg is of type **INT**. But look at the following session excerpt:

```
Use QQ[x];
J:=(7*x)/x;
Type(AsRAT(J));
--> RAT

Use ZZ[x];
K:=(7*x)/x;
Type(AsRAT(K));
--> RAT
```

The question arose from testing the GCD fn:

```
gcd([2,4,AsRAT(6)]);
--> 2   (a RINGELEM in ZZ   <-- **unexpected**)
gcd([2,4,AsRAT(6/5)]);
--> 1   (a RINGELEM in QQ)
```

Related issues:

Related to CoCoA-5 - Bug #110: Surprise return type for GCD of a list of ints	**Closed**	**19 Mar 2012**

History

#1 - 15 May 2012 14:51 - John Abbott

It seems reasonable to me that **AsRAT** give either error or a RAT as result -- after all it is a sort of "cast". The rule should generalize to other AsXXX "casting" fns.

JAA notes that the fns which test whether a value is convertible to RAT (or INT) have rather different names: *viz.* **IsRational** and **IsInteger**. These names are compatible with CoCoALib but rather incompatible with AsRAT and AsINT.

#2 - 15 May 2012 14:56 - John Abbott

AsRAT is defined at lines 31--39 of the file NotBuiltin.cpkg5.
[updating the defn should be pretty easy]

BUT JAA had expected to find it implemented in C++; is this planned?

#3 - 15 May 2012 15:04 - John Abbott

JAA was wondering how to convert a value of type INT into the equivalent value of type RAT. Recall that arithmetic between values of type RAT tries to be smart, and will give a result of type INT if the result is actually integer.

```
AsRAT(3*one(QQ));
3;  --->  type is RAT
AsRAT(3*one(ZZ));
3;  --->  type is RAT
```

#4 - 15 May 2012 15:16 - John Abbott

JAA has modified the defn of **AsRAT** so that it returns a RAT even if the arg given was an INT. All the CoCoA-5 tests pass (but they also all passed when JAA had given a wrong defn).

#5 - 15 May 2012 17:41 - Anna Maria Bigatti

John Abbott wrote:

> It seems reasonable to me that **AsRAT** give either error or a RAT as result -- after all it is a sort of "cast". The rule should generalize to other AsXXX "casting" fns.

I agree

> JAA notes that the fns which test whether a value is convertible to RAT (or INT) have rather different names: *viz.* **IsRational** and **IsInteger**. These names are compatible with CoCoALib but rather incompatible with AsRAT and AsINT.

hmmm, following the rule for **IsXXX**/*AsXXX* those function (in CoCoALib) should be called **IsBigInt** and **IsBigRat** -- that would make it easier to remember the type of the first argument.

On the other hand, the other functions **IsXXX** usually (always?) mean *is implemented as XXX* -- quite different from *it can be cast to a BigInt*.
So maybe it is a good idea it does not follow the rule... but **AsINT**/**AsRAT** are very good names.

.... this decision is more difficult than I expected....

#6 - 15 May 2012 21:13 - John Abbott

- Status changed from New to Resolved

- Assignee set to John Abbott

- % Done changed from 0 to 70

JAA has checked in the modified defn of **AsRAT**.

The question of the names of AsRAT and IsRational remains open.

#7 - 15 May 2012 22:10 - John Abbott

JAA cannot find the manual page for **AsRAT** (ditto **AsINT**).
The only occurrence in CoCoAHelp.xml is in a <see_also>.

#8 - 15 May 2012 22:39 - Anna Maria Bigatti

- Category set to CoCoA-5 function: new

- Target version set to CoCoA-5.0.3

#9 - 30 May 2012 16:40 - John Abbott

- Status changed from Resolved to Closed

- % Done changed from 70 to 100

Done everything.

#10 - 08 Jun 2012 17:26 - Anna Maria Bigatti

- Estimated time set to 2.75 h