

CoCoA-5 - Feature #451

Port function NewSymbol/NewSymbols to CoCoA-5

28 Feb 2014 15:12 - Anna Maria Bigatti

Status:	In Progress	Start date:	28 Feb 2014
Priority:	Normal	Due date:	
Assignee:		% Done:	10%
Category:	CoCoA-5 function: new	Estimated time:	0.00 hour
Target version:	CoCoA-5.?.?	Spent time:	2.00 hours
Description			
In CoCoALib there are functions for making anonymous symbols (i.e. name of indeterminates). Should we port them to CoCoA-5? They are useful for making temporary polynomial rings (which should not be visible to the final user) ensuring no name conflict for symbols.			

History

#1 - 28 Feb 2014 15:21 - Anna Maria Bigatti

I'm already writing the code (taking some rust off my C++) and I'll wait for JAA's approval ;-)

#2 - 28 Feb 2014 15:51 - Anna Maria Bigatti

- Target version set to CoCoA-5.1.0 Easter14

- % Done changed from 0 to 10

I think I made a fool of myself: we cannot have anonymous symbols in CoCoA because we do not have symbols. (we just use strings for names) That's why what I thought to be a trivial function turned out to be so difficult to compile!

All the same: we should invent a mechanism to allow these temporary rings with anonymous indets. So this task is now a **design** issue. (much more difficult)

#3 - 05 Mar 2014 16:50 - John Abbott

One possible solution is to supply an "impossible name" for those indets we want to be "anonymous".

An impossible name could be:

- an empty string;
- a special string;
- a value which is not a string.

A related question: might we want to create two or more rings using the same "anonymous symbol"? If so, perhaps we could use the special string "#" to mean a new anonymous symbol, and "#[3]" to recall a pre-existing anonymous symbol (gives error if the index has not yet been issued). This syntax is similar to the syntax used when printing anonymous symbols.

#4 - 09 Apr 2014 15:53 - John Abbott

- Target version changed from CoCoA-5.1.0 Easter14 to CoCoA-5.?.?

#5 - 09 Nov 2015 15:29 - John Abbott

- Status changed from New to In Progress

Recall that the **use** command automatically sets CoCoA-5 variables whose names coincide with the indets. What will happen if some of the indets are "anonymous"?

Assuming an "anonymous" indet has a "strange" name, the standard mechanism of setting CoCoA-5 variables will not work (since a variable cannot have a "strange" name). This may not matter too much, since the setting of the CoCoA-5 variables is simply a convenience for the user; the anon indet can always be accessed via the `indet(.)` function, though it might be a little tedious determining precisely which index to give.

Maybe there should be a function which returns a list of the indices of the anon indets? (empty list if there are none?)