

CoCoA-5 - Feature #1271

Allow Interruption of printing?

16 Apr 2019 17:43 - John Abbott

Status:	In Progress	Start date:	16 Apr 2019
Priority:	Normal	Due date:	
Assignee:		% Done:	10%
Category:	enhancing/improving	Estimated time:	0.00 hour
Target version:	CoCoA-5.?.?	Spent time:	0.80 hour
Description			
Inside emacs, if you print out a very long list then it takes ages before you can use emacs/CoCoA-5 again. Consider allowing interruption of printing (at least for potentially large objects).			
Related issues:			
Related to CoCoA-5 - Design #610: Variable lt: assign before or after printing		New	02 Sep 2014

History

#1 - 16 Apr 2019 17:49 - John Abbott

I have a tried a first approach, but it did no work as I would have liked...

What I tried:

- in Interpreter.C around line 2453
- inserted CoCoA::CheckForInterrupt("Printing LIST");
- effect: interrupt was recognized, and stopped printing, but resulted in a "catastrophic error" (process exited with code 1) after printing
ERROR UNCAUGHT UNKNOWN EXCEPTION
- printed error seems to have come from Main.C around line 354

Not sure why the exception was not caught.

#2 - 17 Apr 2019 08:26 - Anna Maria Bigatti

I agree this would be very useful.
Strange for the error!

#3 - 17 Apr 2019 10:17 - John Abbott

- *Description updated*
- *Status changed from New to In Progress*
- *% Done changed from 0 to 10*

Which printing commands should allow themselves to be interrupted?

- lists
- matrices
- ideals (?)

I think probably not polynomials (nor very large integers):

- for large integers it would be tricky because they are internally converted to a string, which is then printed with a single C++ fn call
- for polynomials we could perhaps check for interrupts between printing terms, but I am concerned that too many checks for interrupts might have

a perceptible impact on speed (though the checks should be quite cheap); also note that polynomials can have a "recursive structure".

I had thought about checking for interrupts, say, every 100 terms in a poly. But then it is possible to create a large recursive polynomial where no check is made: consider $(1+x+x^2+\dots+x^{99})(1+y+y^2+\dots+y^{99})(1+z+z^2+\dots+z^{99})$ in the ring $\mathbb{Q}\mathbb{Q}[x][y][z]$.

#4 - 04 Mar 2020 22:23 - John Abbott

- Related to Design #610: Variable *It*: assign before or after printing added