

CoCoALib - Bug #1345

Problem with linking the library

19 Oct 2019 11:43 - Anna Maria Bigatti

Status:	Closed	Start date:	19 Oct 2019
Priority:	High	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	Safety	Estimated time:	1.11 hour
Target version:	CoCoALib-0.99650 November 2019	Spent time:	1.20 hour
Description			
<p>I was testing a change made in SparsePolyOps-ideal-monomial.C, and, though it did recompile the file, the new code is not run (I even changed it to return ideal(one(R))!).</p> <p>make clean; make</p> <p>apparently solved it, but now, wanting to remove the debug printing, the problem resurfaced: it compiles, but the new code is not used.</p> <p>What's going on??????</p> <p>MacOS, clang.</p>			

History

#1 - 19 Oct 2019 17:14 - Anna Maria Bigatti

- Status changed from New to Closed
- Assignee set to Anna Maria Bigatti
- % Done changed from 10 to 100

My mistake?

I tried again, and I cannot longer reproduce it.
It is linked correctly.

#2 - 19 Oct 2019 19:43 - John Abbott

- Estimated time set to 1.11 h

Very strange. I think I may have seen a similar phenomenon occasionally, but make clean; make always seemed to fix it for me.

Anyway, it is good to note that it has happened.

If something like it happens again, try using `ls -l` on the source file, on the file.o, and on libcocoa.a just to make sure that all files have been changed after you changed the source. Perhaps you ran make inside `src/AlgebraicCore/`, but did not update the library? Perhaps you were working in two separate copies of the source tree?

If you were running your tests from a cocoa5 session, then you should also check with `ls -l` that CoCoAInterpreter has been updated. (maybe `@VersionInfo()` will tell you?)

#3 - 19 Oct 2019 20:28 - Anna Maria Bigatti

John Abbott wrote:

If you were running your tests from a cocoa5 session, then you should also check with `ls -l` that `CoCoAInterpreter` has been updated.
(maybe `@VersionInfo()` will tell you?)

Yes, I did check with `VersionInfo()`, and I did see the file being compiled.
That's why I thought it was a problem with linking.

Anyway, it works now.

I guess I had misinterpreted a printing message of mine in my second try after `make clean; make`, because now it compiles and links as expected.