CoCoALib - Feature #82

C++11 compatibility questions

26 Jan 2012 14:58 - John Abbott

Status:	Closed	**Start date:**	26 Jan 2012
Priority:	Normal	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	Portability	**Estimated time:**	1.99 hour
Target version:	CoCoALib-0.99850	**Spent time:**	1.80 hour

Description

THIS HAS BEEN SUPERSEDED BY ISSUE #1225

When CoCoALib is converted to C++11 several aspects of the code will have to be reconsidered.
The purpose of this task is simply to gather a list of these aspects.

Related issues:

Related to CoCoA-5 - Design #83: C++11 compatibility questions	**In Progress**	**26 Jan 2012**
Related to CoCoALib - Design #581: C++14: MachineInt	**Closed**	**04 Jul 2014**
Related to CoCoALib - Feature #1010: C++11: Mario's Hilbert scheme code	**New**	**20 Feb 2017**
Related to CoCoALib - Design #710: Update normaliz interface?	**In Progress**	**17 May 2015**
Related to CoCoALib - Design #999: configuration: include -std=c++03 by default?	**Closed**	**18 Jan 2017**
Related to CoCoALib - Support #887: My first compilations with clang	**Closed**	**31 May 2016**
Related to CoCoALib - Design #1166: C++11: allows large integer literals (in ...	**Closed**	**12 Mar 2018**
Related to CoCoALib - Design #894: strict enum types: C++11 extension	**In Progress**	**21 Jun 2016**
Related to CoCoALib - Design #891: Replace auto_ptr in preparation for C++11?	**Closed**	**17 Jun 2016**
Related to CoCoALib - Bug #867: Compilation Error: C++11 and old GMP-Version	**Closed**	**13 Apr 2016**
Related to CoCoALib - Support #861: Janet basis code: TmpJB files give some p...	**Closed**	**31 Mar 2016**
Related to CoCoALib - Design #1225: Move to C++14 (skipping C++11)	**In Progress**	**06 Sep 2018**
Related to CoCoALib - Design #1242: C++14: Use type auto where appropriate	**In Progress**	**08 Feb 2019**

History

#1 - 28 Jan 2013 08:05 - Anna Maria Bigatti

- Category set to Portability

#2 - 01 Apr 2014 17:35 - Anna Maria Bigatti

- Target version set to CoCoALib-0.99533 Easter14

#3 - 07 Apr 2014 14:43 - John Abbott

- Target version changed from CoCoALib-0.99533 Easter14 to CoCoALib-1.0

Probably several "efficient" procedures can be eliminated.

- add, sub, mul, div for RingElem
- add "move" style ctors to many types
- eliminate CoCoAVector (see issue #357)

#4 - 28 Jun 2015 16:38 - John Abbott

It seems that a few things noted in issue #83 should actually be in this issue.

Apparently C++11 has a way of finding out the name of the function you are in: __func__ behaves like a local variable (of type const char* const). This may be useful when calling CoCoA_ERROR, but note that the standard does not say much about the actual value of the string (so our current

manual approach is possibly safer, or at least consistent across different platforms).

#5 - 19 Apr 2016 14:27 - John Abbott

- Status changed from New to In Progress

- % Done changed from 0 to 10

Mario has found a web page which explains that the preprocessor flag __cpluscplus has a numerical value indicating which version of C++ is supported by the compiler. This appears to be an official part of C++ (but perhaps there is some doubt as to how well it is supported by compilers).

Old C++ corresponds to the value 199711L -- L is needed in case preprocessor is 16bit (I guess).
g++ with the flag -std=c++11 gives the value 201103.

This suggests that we can do something like:

```
#if __cplusplus < 201100L
 // old code
#else
 // new code
#endif
```

NOTE I have just made a modification to debug_new.C and it appears to have worked as hoped. :-)

#6 - 20 Feb 2017 13:22 - John Abbott

- Related to Feature #1010: C++11: Mario's Hilbert scheme code added

#7 - 20 Feb 2017 13:23 - John Abbott

How does this issue differ from issue #83?

#8 - 06 Nov 2017 13:43 - John Abbott

- Related to Design #710: Update normaliz interface? added

#9 - 08 Nov 2017 17:08 - John Abbott

- Related to Design #999: configuration: include -std=c++03 by default? added

#10 - 16 Dec 2017 15:16 - John Abbott

- Related to Support #887: My first compilations with clang added

#11 - 12 Mar 2018 11:48 - John Abbott

- Related to Design #1166: C++11: allows large integer literals (in NumTheory-prime.C) added

#12 - 26 Jun 2018 15:21 - John Abbott

- Related to Design #894: strict enum types: C++11 extension added

#13 - 26 Jun 2018 15:21 - John Abbott

- Related to Design #891: Replace auto_ptr in preparation for C++11? added

#14 - 26 Jun 2018 15:21 - John Abbott

- Related to Bug #867: Compilation Error: C++11 and old GMP-Version added

#15 - 26 Jun 2018 15:21 - John Abbott

- Related to Support #861: Janet basis code: TmpJB files give some problems with C++11 (using CLANG/LLVM) added

#16 - 26 Jun 2018 15:25 - John Abbott

I read somewhere on the internet that it is probably better to jump straight to C++14 rather than to C++11; the argument was that C++14 tied up various loose ends which C++11 had. I do not recall the source, but believe it was reliable.

#17 - 06 Sep 2018 16:15 - John Abbott

- Related to Design #1225: Move to C++14 (skipping C++11) added

#18 - 08 Feb 2019 21:36 - John Abbott

- Related to Design #1242: C++14: Use type auto where appropriate added

#19 - 07 Mar 2019 09:48 - Anna Maria Bigatti

Warning with clang:

```
toric.C:2551:3: warning: 'register' storage class specifier is deprecated and
      incompatible with C++1z [-Wdeprecated-register]
  register  int n, i, BPLen, BLen;
  ^~~~~~~~~
```

#20 - 19 Mar 2021 13:27 - John Abbott

- Target version changed from CoCoALib-1.0 to CoCoALib-0.99850

- % Done changed from 10 to 20

Do we have progress to report?

#21 - 21 Jun 2021 15:41 - Anna Maria Bigatti

John Abbott wrote:

> Do we have progress to report?

removed (unused) keyword register from toric.C

#22 - 19 Jul 2021 13:49 - John Abbott

- Description updated

#23 - 18 Feb 2022 15:23 - John Abbott

- % Done changed from 20 to 60

#24 - 26 Sep 2022 20:34 - John Abbott

- Status changed from In Progress to Closed

- Assignee set to John Abbott

- % Done changed from 60 to 100

- Estimated time set to 1.99 h

There is a big bold comment in the main descr saying that this issue has been superseded by #1225.
For this reason I am closing this issue; also I am not sure how it differs from #82