CoCoALib - Feature #313

Elim(vector<long>) as PPOrderingCtor

15 Feb 2013 17:13 - Anna Maria Bigatti

Status:	In Progress	**Start date:**	15 Feb 2013
Priority:	High	**Due date:**	
Assignee:	Anna Maria Bigatti	**% Done:**	30%
Category:	New Function	**Estimated time:**	0.00 hour
Target version:	CoCoALib-0.99880	**Spent time:**	0.35 hour

Description

Same as lex, StdDegLex, ... also **Elim(vector<long>)** should be implemented as a **PPOrderingCtor**

Related issues:

Related to CoCoA-5 - Feature #78: Elim ordering and matrix ordering in CoCoA-5	**In Progress**	**16 Jan 2012**
Related to CoCoALib - Design #822: Should ElimMat return a ConstMatrix	**Closed**	**25 Nov 2015**
Related to CoCoALib - Feature #1808: New constructor for PolyRing with elimin...	**In Progress**	**28 Mar 2024**

History

#1 - 29 Oct 2013 14:29 - Anna Maria Bigatti

- Target version changed from CoCoALib-0.9953 to CoCoALib-0.99532

#2 - 24 Jan 2014 20:40 - John Abbott

- Priority changed from Normal to High

Anna told me she has an impl (of exactly what I don't recall).
Let's make it available -- I'd like it to be available in C5 too.

For what it's worth, here's my impl in C5 language

```
Define ElimOrdMat(N, ElimIndets)
  // Check entries of ElimIndets are (distinct) ints in range 1..N
  TopLevel ZZ;
  M := NewMat(N,N,0);
  Foreach i In ElimIndets Do
    M[1,i] := 1;
  EndForeach;
  E := len(ElimIndets);
  For i := 2 To E Do
    M[i, ElimIndets[E+2-i]] := -1;
  EndFor;

  KeepIndets := [k | k In 1..N And Not(k IsIn ElimIndets)];
  Foreach j In KeepIndets Do
    M[E+1,j] := 1;
  EndForeach;
  K := len(KeepIndets);
  For i := 2 To len(KeepIndets) Do
    M[E+i, KeepIndets[K+2-i]] := -1;
  EndFor;
  Return M;
EndDefine; -- ElimOrdMat
```

#3 - 01 Apr 2014 19:26 - Anna Maria Bigatti

- Target version changed from CoCoALib-0.99532 to CoCoALib-0.99533 Easter14

#4 - 07 Apr 2014 14:37 - John Abbott

- Target version changed from CoCoALib-0.99533 Easter14 to CoCoALib-0.99534 Seoul14

#5 - 11 Jul 2014 12:07 - John Abbott

- Target version changed from CoCoALib-0.99534 Seoul14 to CoCoALib-1.0

#6 - 30 Sep 2016 13:19 - John Abbott

- Target version changed from CoCoALib-1.0 to CoCoALib-0.99550 spring 2017

#7 - 30 Sep 2016 13:30 - John Abbott

- Related to Design #822: Should ElimMat return a ConstMatrix added

#8 - 28 Mar 2017 17:11 - John Abbott

Should we postpone this issue to the next version?

#9 - 28 Mar 2017 21:51 - Anna Maria Bigatti

- Target version changed from CoCoALib-0.99550 spring 2017 to CoCoALib-0.99560

#10 - 06 Nov 2017 13:58 - John Abbott

- Target version changed from CoCoALib-0.99560 to CoCoALib-0.99600

#11 - 31 Jul 2018 14:24 - Anna Maria Bigatti

- Target version changed from CoCoALib-0.99600 to CoCoALib-0.99650 November 2019

#12 - 23 Sep 2019 12:05 - John Abbott

- Target version changed from CoCoALib-0.99650 November 2019 to CoCoALib-0.99700

#13 - 09 Jan 2020 12:16 - John Abbott

- Target version changed from CoCoALib-0.99700 to CoCoALib-0.99800

#14 - 05 Nov 2021 15:58 - Anna Maria Bigatti

- % Done changed from 0 to 30

#15 - 28 Jan 2022 13:05 - John Abbott

- Target version changed from CoCoALib-0.99800 to CoCoALib-0.99850

#16 - 08 Mar 2024 17:40 - John Abbott

- Assignee set to Anna Maria Bigatti

#17 - 22 Mar 2024 18:17 - Anna Maria Bigatti

- Status changed from New to In Progress

- Target version changed from CoCoALib-0.99850 to CoCoALib-0.99880

Probably the functions ElimMat-like are good enough for this.
At least we should write some explicit example here (for CoCoA-5 #78, and for CoCoALib)

#18 - 28 Mar 2024 07:36 - Anna Maria Bigatti

- Related to Feature #1808: New constructor for PolyRing with elimination ordering added