cyclic9-verif

```
n=9;R=PolynomialRing(QQ,n,'x');
I = sage.rings.ideal.Cyclic(R,n);
I.gens()
```

```
    [x0 + x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8, x0*x1 + x1*x2 + x2*x3 +
    x3*x4 + x4*x5 + x5*x6 + x6*x7 + x0*x8 + x7*x8, x0*x1*x2 + x1*x2*x3 +
    x2*x3*x4 + x3*x4*x5 + x4*x5*x6 + x5*x6*x7 + x0*x1*x8 + x0*x7*x8 +
    x6*x7*x8, x0*x1*x2*x3 + x1*x2*x3*x4 + x2*x3*x4*x5 + x3*x4*x5*x6 +
    x4*x5*x6*x7 + x0*x1*x2*x8 + x0*x1*x7*x8 + x0*x6*x7*x8 + x5*x6*x7*x8,
    x0*x1*x2*x3*x4 + x1*x2*x3*x4*x5 + x2*x3*x4*x5*x6 + x3*x4*x5*x6*x7 +
    x0*x1*x2*x3*x8 + x0*x1*x2*x7*x8 + x0*x1*x6*x7*x8 + x0*x5*x6*x7*x8 +
    x4*x5*x6*x7*x8, x0*x1*x2*x3*x4*x5 + x1*x2*x3*x4*x5*x6 +
    x2*x3*x4*x5*x6*x7 + x0*x1*x2*x3*x4*x8 + x0*x1*x2*x3*x7*x8 +
    x0*x1*x2*x6*x7*x8 + x0*x1*x5*x6*x7*x8 + x0*x4*x5*x6*x7*x8 +
    x3*x4*x5*x6*x7*x8, x0*x1*x2*x3*x4*x5*x6 + x1*x2*x3*x4*x5*x6*x7 +
    x0*x1*x2*x3*x4*x5*x8 + x0*x1*x2*x3*x4*x7*x8 + x0*x1*x2*x3*x6*x7*x8 +
    x0*x1*x2*x5*x6*x7*x8 + x0*x1*x4*x5*x6*x7*x8 + x0*x3*x4*x5*x6*x7*x8 +
    x2*x3*x4*x5*x6*x7*x8, x0*x1*x2*x3*x4*x5*x6*x7 +
    x0*x1*x2*x3*x4*x5*x6*x8 + x0*x1*x2*x3*x4*x5*x7*x8 +
    x0*x1*x2*x3*x4*x6*x7*x8 + x0*x1*x2*x3*x5*x6*x7*x8 +
    x0*x1*x2*x4*x5*x6*x7*x8 + x0*x1*x3*x4*x5*x6*x7*x8 +
    x0*x2*x3*x4*x5*x6*x7*x8 + x1*x2*x3*x4*x5*x6*x7*x8,
    x0*x1*x2*x3*x4*x5*x6*x7*x8 - 1]
```

```
%magma
R<x0,x1,x2,x3,x4,x5,x6,x7,x8> := PolynomialRing( RationalField(),9, "grevlex" );
```

```
magma.eval("I:=Ideal(%s);"%(I.gens()));
```

```
    ''
```

```
%magma
I;
```

```
    Ideal of Polynomial ring of rank 9 over Rational Field
    Order: Graded Reverse Lexicographical
    Variables: x0, x1, x2, x3, x4, x5, x6, x7, x8
    Basis:
    [
    x0 + x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8,
    x0*x1 + x1*x2 + x2*x3 + x3*x4 + x4*x5 + x5*x6 + x6*x7 + x0*x8 +
    x7*x8,
    x0*x1*x2 + x1*x2*x3 + x2*x3*x4 + x3*x4*x5 + x4*x5*x6 + x5*x6*x7 +
    x0*x1*x8 + x0*x7*x8 + x6*x7*x8,
    x0*x1*x2*x3 + x1*x2*x3*x4 + x2*x3*x4*x5 + x3*x4*x5*x6 + x4*x5*x6*x7
    + x0*x1*x2*x8 + x0*x1*x7*x8 + x0*x6*x7*x8 + x5*x6*x7*x8,
    x0*x1*x2*x3*x4 + x1*x2*x3*x4*x5 + x2*x3*x4*x5*x6 + x3*x4*x5*x6*x7 +
    x0*x1*x2*x3*x8 + x0*x1*x2*x7*x8 + x0*x1*x6*x7*x8 + x0*x5*x6*x7*x8 +
    x4*x5*x6*x7*x8,
    x0*x1*x2*x3*x4*x5 + x1*x2*x3*x4*x5*x6 + x2*x3*x4*x5*x6*x7 +
    x0*x1*x2*x3*x4*x8 + x0*x1*x2*x3*x7*x8 + x0*x1*x2*x6*x7*x8 +
    x0*x1*x5*x6*x7*x8 + x0*x4*x5*x6*x7*x8 + x3*x4*x5*x6*x7*x8,
    x0*x1*x2*x3*x4*x5*x6 + x1*x2*x3*x4*x5*x6*x7 + x0*x1*x2*x3*x4*x5*x8 +
    x0*x1*x2*x3*x4*x7*x8 + x0*x1*x2*x3*x6*x7*x8 + x0*x1*x2*x5*x6*x7*x8 +
    x0*x1*x4*x5*x6*x7*x8 + x0*x3*x4*x5*x6*x7*x8 + x2*x3*x4*x5*x6*x7*x8,
    x0*x1*x2*x3*x4*x5*x6*x7 + x0*x1*x2*x3*x4*x5*x6*x8 +
    x0*x1*x2*x3*x4*x5*x7*x8 + x0*x1*x2*x3*x4*x6*x7*x8 +
    x0*x1*x2*x3*x5*x6*x7*x8 + x0*x1*x2*x4*x5*x6*x7*x8 +
    x0*x1*x3*x4*x5*x6*x7*x8 + x0*x2*x3*x4*x5*x6*x7*x8 +
    x1*x2*x3*x4*x5*x6*x7*x8,
    x0*x1*x2*x3*x4*x5*x6*x7*x8 - 1
    ]
```

```
from giacpy import *
```

```
    // Giac share root-directory:/home/han/dev/sage/local/share/giac/
    // Unable to find keyword file
    /home/han/dev/sage/local/share/giac/doc/fr/keywords
    // Giac share root-directory:/home/han/dev/sage/local/share/giac/
    Help file /home/han/dev/sage/local/share/giac/doc/fr/aide_cas not
    found
    Added 0 synonyms
```

```
giacsettings.threads=8; # maximum number of threads allowed
giacsettings.proba_epsilon=1e-7; # for probabilistic algorithms
```

```
time BGbis=loadgiacgen('/home/han/cyclic9.giacgen')
```

```
    Time: CPU 68.20 s, Wall: 68.34 s
```

```
Br=BGbis.revlist();Br.dim()
```

```
    1344
```

```
%magma
time B:=GroebnerBasis(I);
```

```
    Time: 29048.530
```

```
tutu=libgiac(magma('B[1]'))
```

```
(Br[0]/tutu).normal().type()=='DOM_INT'
```

```
    True
```

Compare the result from magma with the result loaded from giacpy

```
%time
T=True;
for k in range(len(Br)):
    TT=(Br[k]/libgiac(magma("B[%s]"%(k+1)))).normal().type()=='DOM_INT'
    if (not TT):
        print(k)
```

```
    T=T and TT

T
```

```
    True
    CPU time: 1278.72 s,  Wall time: 1332.52 s
```

```
MR=magma('R');MR
```

```
    Polynomial ring of rank 9 over Rational Field
    Order: Graded Reverse Lexicographical
    Variables: x0, x1, x2, x3, x4, x5, x6, x7, x8
```

```
time IM=MR.ideal("B")
```

```
    Time: CPU 0.00 s, Wall: 0.10 s
```

```
IM.Dimension()
```

```
    2
```

```
1
```