EXAMPLE FILE FOR M2INTEX

PAUL ZINN-JUSTIN

1. Introduction

some basic examples:

```
i1 : R=QQ[x,y]; factor(x^3-y^3)
```

$$o2 = (x-y)\,(x^2+x\,y+y^2)$$

```
o2 : Expression of class Product
i3 : res coker vars R
```

$$o3 = R^1 \xleftarrow{\begin{pmatrix} x & y \end{pmatrix}} R^2 \xleftarrow{\begin{pmatrix} -y \\ x \end{pmatrix}} R^1 \xleftarrow{0} 0$$
$$\qquad\quad 0 \qquad\qquad 1 \qquad\qquad 2 \qquad 3$$

```
o3 : ChainComplex
i4 : OO_(Proj(R/(x^3-y^3)))^{1,2}
```

$$o4 = \mathcal{O}^1_{\mathtt{Proj}\left(\frac{R}{x^3-y^3}\right)}(1) \;\oplus\; \mathcal{O}^1_{\mathtt{Proj}\left(\frac{R}{x^3-y^3}\right)}(2)$$

```
o4 : coherent sheaf on Proj (R/(x^3-y^3)), free
```

more:

```
i5 : 318/46
```

$$o5 = \frac{159}{23}$$

```
o5 : QQ
i6 : exp 3.73767
```

$$o6 = 42$$

```
o6 : RR (of precision 53)
```

strings:

```
i7 : "hehe"
o7 = hehe
```

and nets:

```
i8 : "haha"||"hoho"
o8 =  haha
      hoho
```

printing:

```
i9 : for i from 1 to 8 do print(i^i)
1
4
27
256
3125
46656
823543
16777216
```

2. Help

```
i10 : help det
o10 =
```

determinant -- determinant of a matrix

Synopsis

- Usage:
  ```
  det M
  ```
- Inputs:
  ```
  -- M, a square matrix
  ```
- Optional inputs:
  ```
  -- Strategy => ..., default value null, choose between Bareiss and Cofactor
     algorithms
  ```
- Outputs:
  ```
  -- a ring element, which is the determinant of M
  ```

Description

See also

- exteriorPower -- exterior power
- minors -- ideal generated by minors
- permanents -- ideal generated by square permanents of a matrix
- pfaffians -- ideal generated by Pfaffians

Ways to use determinant :

- "determinant(Matrix)"
- "determinant(MutableMatrix)"

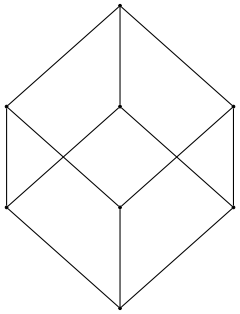
For the programmer

```
The object determinant is a method function with options.
o10 : DIV
```

3. Packages

packages that have a tex output will work:

```
i11 : needsPackage "Posets";
i12 : booleanLattice 3
```



```
o12 =
o12 : Poset
```

4. Tricky examples

```
i13 : -- some tricky examples
```

A bunch of complicated cases: a multi-line example

```
      f = i -> (
      -- that's dumb
      i+1
      )
o13 = f
o13 : FunctionClosure
```

and another weirder one:

```
i14 : I=ideal 0; f = i -> (
o14 : Ideal of ℤ
      i+1)
o15 = f
o15 : FunctionClosure
```

finally:

```
i16 : a=1;b=2;
i18 : c=3;
```

That last one has no output.