1 NfOrder: Orders in in absolute number fields

1.1 Introduction

Assume that K is an algebraic number field of degree d. Classically, an order $\mathcal{O}$ of K is a subring $\mathcal{O} \subseteq K$, which is a free $\mathbf{Z}$-module of rank $d = \operatorname{degree}(K)$. For computational reasons, an $\mathcal{O}$ is always given together with a matrix A (the *basis matrix*), such that the elements $\omega_1, \dots, \omega_d$ defined by

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_d \end{pmatrix} = A \cdot \begin{pmatrix} 1 \\ \alpha \\ \vdots \\ \alpha^d \end{pmatrix}$$

are a $\mathbf{Z}$-basis of $\mathcal{O}$.

In HECKE orders in number fields are modelled by objects of type `NfOrder`.

1.2 Creation of objects of type `NfOrder`

```
EquationOrder(K::NfNumberField) -> NfOrder
    arg1::Bool = true
    I am a paramter::Bool = false
```

Assume that `K` represents the number field $\mathbf{Q}(\alpha)$. This function will produce the equation order $\mathbf{Z}[\alpha]$ of α, that is, the order of K with basis matrix $\mathbf{1}_d$.

```
I'm an example
  julia> Qx, x = PolynomialRing(QQ,"x");
  julia> f = x^19 + x^2 + 3*x + 11;
  julia> K, a = NumberField(f, "a");
  julia> O = EquationOrder(K);
```

```
Order(a) -> b
```

```
Order(arg1) -> tar
    parp
```

ich bin eine funkction. Die beschreibung kann so lang sein wie sie will. blablabla. dasdasdksadkasdksadkasdka sadk sakdk askdas .

1.3 Creation of elements

```
O(x::nf_elem) -> NfOrderEllem
    check::Bool = true
```

Given an order `O`, try to coerce the number field element `x` into `O`

```
I'm the example
```
