Das Paket JanetOre ist eine Variante von Involutive, welches mit diversen iterierten Schiefpolynomringen arbeitet.

```
> with(JanetOre);
```

[*AssertJBasis, JAddRhs, JAnnihilator, JBasis, JBasisFast, JCartanCharacter, JCheckHom, JCoeff, JCoeffList, JCokernel, JDefect, JDerMod, JDimension, JDirectSum, JEulerChar, JExt1, JExtn, JFactorModuleBasis, JFactorize, JGroebnerBasis, JGroebnerBasisFast, JHF, JHP, JHas, JHilbertFunction, JHilbertPolynomial, JHilbertSeries, JHom, JHomHom, JIndexRegularity, JIntersection, JInvReduce, JInvReduceFast, JInvolution, JKaehler, JKernel, JLeadingMonomial, JLeftInverse, JMinPoly, JMult, JNotHas, JParametrization, JPrincDeriv, JRepres, JResolution, JResolutionDim, JRightDivide, JRightInverse, JSubFactor, JSum, JSyzOp, JSyzygies, JSyzygyModule, JSyzygyModuleFast, JTabVar, JTorsion, JWeightedHilbertSeries, JZeroSets, JanetOreOptions, JanetOreStats*]

```
> var:=[delta[1],delta[2],t[1],t[2]];
```

$$var := [\delta_1, \delta_2, t_1, t_2]$$

```
> ops:=[weyl(delta[1],t[1]),weyl(delta[2],t[2])];
```

$$ops := [\mathrm{weyl}(\delta_1, t_1), \mathrm{weyl}(\delta_2, t_2)]$$

```
> L:=expand([(t[1]+1)*(delta[2]-1),t[1]*delta[1]^2-t[2]^2]);
```

$$L := [t_1\,\delta_2 - t_1 + \delta_2 - 1,\ t_1\,\delta_1^{\,2} - t_2^{\,2}]$$

```
> JBasis(L,var,ops);
```

$$[1]$$

Der folgende Befehl, dessen Output wir wegen seiner Länge nicht abdrucken, liefert die beiden Koeffizienten, mit denen man die beiden Elemente von L linearkombinieren muß, um 2 zu erhalten

```
> JBasis(JAddRhs(L),var,ops):
```

Wir überprüfen unser Ergebnis, indem wir den Janetalgorithmus in dem größeren Ring benutzen.

```
> with(Janet):
> ivar:=[x,y];dvar:=[u];
```

$$ivar := [x, y]$$

$$dvar := [u]$$

```
> l:=Ind2Diff([x*u[y]-x*u+u[y]-u-a(x,y),x*u[x,x]-y^2*u-b(x,y)],iva
  r,dvar);
```

$$l := \left[x\left(\frac{\partial}{\partial y}\mathrm{u}(x,y)\right) - x\,\mathrm{u}(x,y) + \left(\frac{\partial}{\partial y}\mathrm{u}(x,y)\right) - \mathrm{u}(x,y) - \mathrm{a}(x,y),\ x\left(\frac{\partial^2}{\partial x^2}\mathrm{u}(x,y)\right) - y^2\,\mathrm{u}(x,y) - \mathrm{b}(x,y)\right]$$

```
> JanetBasis(l,ivar,dvar);
```

$$\left[\left[\mathrm{u}(x,y) - \frac{1}{2}\left(\left(\frac{\partial^2}{\partial x^2}\mathrm{a}(x,y)\right)x + 2\left(\frac{\partial^2}{\partial x^2}\mathrm{a}(x,y)\right)x^2 + \left(\frac{\partial^2}{\partial x^2}\mathrm{a}(x,y)\right)x^3 - 2\left(\frac{\partial}{\partial x}\mathrm{a}(x,y)\right)x\right.\right.$$

$$\left. - 2\left(\frac{\partial}{\partial x}\mathrm{a}(x,y)\right)x^2 + 2\,\mathrm{a}(x,y)\,x - \left(\frac{\partial}{\partial y}\mathrm{b}(x,y)\right) - 3\left(\frac{\partial}{\partial y}\mathrm{b}(x,y)\right)x - 3\left(\frac{\partial}{\partial y}\mathrm{b}(x,y)\right)x^2 - \left(\frac{\partial}{\partial y}\mathrm{b}(x,y)\right)x^3\right.$$

$$\left.\left. + \mathrm{b}(x,y) + 3\,\mathrm{b}(x,y)\,x + 3\,\mathrm{b}(x,y)\,x^2 + \mathrm{b}(x,y)\,x^3 - y^2\,\mathrm{a}(x,y) - 2\,y^2\,\mathrm{a}(x,y)\,x - y^2\,\mathrm{a}(x,y)\,x^2\right) \Big/ \left(\right.\right.$$

$$(1+x)^3\, y)\Big],\ [x, y],\ [u]\Big]$$

Setzt man a=b=0, so hat man eine Bestätigung des oberen Ergebnisses. Man beachte, daß die Linearkombination, die 1 liefert, wesentlich kompaktere Koeffizienten hat. Dies bestätigt nochmals daß wir oben ein viel genaueres Ergebnis berechnet haben.

Wir wollen die obigen Daten als im Translations- oder Differenzenring benutzen.

```
> ops:=[shift(delta[1],t[1]),shift(delta[2],t[2])];
```

$$ops := [\mathrm{shift}(\delta_1, t_1), \mathrm{shift}(\delta_2, t_2)]$$

```
> JBasis(L,var,ops);
```

$$[t_1 + 1,\ t_2^{\,2},\ t_1\,\delta_1 + 2\,\delta_1,\ \delta_1^{\,2},\ \delta_1\,t_2^{\,2}]$$

```
> JFactorModuleBasis(var);
```

$$\frac{\delta_1}{1-\delta_2} + \frac{\delta_1\, t_2}{1-\delta_2} + \frac{1}{1-\delta_2} + \frac{t_2}{1-\delta_2}$$

```
> JHilbertSeries(t);
```

$$1 + 3\,t + 4\,t^2 + \frac{4\,t^3}{1-t}$$

```
>
```