This uses software packages by Frederick Chyzak. Follow the installation instructions on the class homepage, or the following commands will not work.

```maple
> restart;
> with(Ore_algebra); with(Groebner); with(Holonomy);
with(Mgfun);

Ore_to_DESol Ore_to_RESol Ore_to_diff Ore_to_shift annihilators applyopr,
diff_algebra poly_algebra qshift_algebra rand_skew_poly shift_algebra,
skew_algebra skew_elim skew_gcdex skew_pdiv skew_power skew_prem,
skew_product

fglm_algo gbasis gsolve hilbertdim hilbertpoly hilbertseries inter_reduce,
is_finite is_solvable leadcoeff leadmon leadterm normalf pretend_gbasis,
reduce spoly termorder testorder univpoly

algeq_to_dfinite dfinite_add dfinite_mul holon_closure holon_defint,
holon_defqsum holon_defsum holon_diagonal hypergeom_to_dfinite,
takayama_algo

[ diag_of_sys, int_of_sys, pol_to_sys, sum_of_sys, sys*sys, sys+sys ]
```

Define a noncommutative algebra with a shift operator

Sn f(n,k,x) = f(n+1,k,x)

and a difference (Delta) operator

Dk f(n,k,x) = f(n,k+1,x)-f(n,k,x)

and a differential operator

Dx f(n,k,x) = d/dx f(n,k,x)

(There are only so many letters available, so he used D to denote these two separate things.)

There are other predefined types of Ore algebras, as well as the ability to define your own.

```maple
> A:=skew_algebra(shift=[Sn,n],delta=[Dk,k],diff=[Dx,x],polyном=k);
```
We can multiply two operators together with skew_product(f,g,A):

```maple
> showprod := proc(f,g,A)
    print(f &* g = skew_product(f,g,A))
end:
> showprod(x,Dx,A);

\[ x \&* D_x = x \, D_x \]

> showprod(Dx,x,A);

\[ D_x \&* x = 1 + x \, D_x \]

> showprod(n,Sn,A); showprod(Sn,n,A);

\[ n \&* S_n = n \, S_n \]

\[ S_n \&* n = (n + 1) \, S_n \]

> showprod(Dx,x^2,A);

\[ D_x \&* x^2 = D_x \, x^2 + 2 \, x \]

**Warning:** Maple understands commutative polynomials, but doesn’t really understand noncommutative ones. Chyzak’s software recognizes this and works around it. Any monomial in \( x, D_x \) (or \( n, S_n \), etc.) that is properly of the form \( x^a \, D_x^b \) may be represented by Maple in either order; that way, or \( D_x^b \, x^a \), but Chyzak’s software assumes it’s intended the \( x \)’s be left and the \( D_x \)’s be right.

> showprod(Sn*Dx,n*x^2,A);

\[ S_n \, D_x \&* n \, x^2 = (2 \, n + 2 \, x) \, S_n + (n \, x^2 + x^2) \, S_n \, D_x \]

We can also apply an operator to a function:

> applyopr(Dx,sin(x),A);

\[ \cos(x) \]

> applyopr(Dx,x,A);

\[ 1 \]

> applyopr(Dx,x^2,A);

\[ 2 \, x \]

Random skew polynomials can be generated (for instance, to create random input for routines):

> rand_skew_poly(x,A);

\[ -85 \, x^5 - 55 \, x^4 - 37 \, x^3 - 35 \, x^2 + 97 \, x + 50 \]

> rand_skew_poly([x,Dx],A);

\[ -8 \, x^5 - 93 \, x^4 + (45 \, x + 43 \, x^4) \, D_x + (92 - 62 \, x^3) \, D_x^2 \]

> rand_skew_poly([x,Dx],terms=5,A);

\[ -61 - 50 \, D_x - 12 \, x^3 - 18 \, D_x^3 + 31 \, x^2 \, D_x^2 \]

**Application.** Find operators in this algebra that annihilate binomial(n,k)
Verify that they do annihilate it. Apply the operators to the function.

```plaintext
> applyopr(el[1], binomial(n, k), A);

[(1 - n + 2 k) binomial(n, k) + (k + 1) (binomial(n, k + 1) - binomial(n, k))]
```

```plaintext
> map(applyopr, el, binomial(n, k), A);

[[0, 0, 0]]
```

**Noncommutative division in K(n)[Sn]**

```plaintext
> A := shift_algebra([Sn, n]);

A := Ore_algebra

> f1 := skew_power((n+1)*Sn, 2, A);

f1 := (n^2 + 3 n + 2) Sn^2

> f2 := skew_product(Sn+5, f1, A) + Sn+9;

f2 := (5 n^2 + 15 n + 10) Sn^2 + (n^2 + 5 n + 6) Sn^3 + Sn + 9

> d1 := skew_pdiv(f2, (n+1)*Sn, Sn, A);

d1 := [n + 1, Sn^2 n^2 + 3 Sn^2 n + 5 Sn n^2 + 10 Sn n + 2 Sn^2 + 5 Sn + 1, 9 n + 9]

> skew_product(d1[1], f2, A) -

skew_product(d1[2], (n+1)*Sn, A);

9 n + 9

> d2 := skew_pdiv(f2, (n+1)*Sn, n, A);

d2 := [1, Sn^2 n + 2 Sn^2 + 5 Sn n + 5 Sn, Sn + 9]

> skew_product(d2[1], f2, A) -

skew_product(d2[2], (n+1)*Sn, A);

Sn + 9
```

**skew_pdiv(p,q,x,A)**  --- >  [ u, v, r ]

where \( u^*p - v^*q = r \) of x-degree lower than q.

v and r are polynomials in x, while u is a coefficient.

```plaintext
> skew_pdiv((n+3)^5, n^2+2, n, A);

[1, n^3 + 15 n^2 + 88 n + 240, 229 n - 237]
```

```plaintext
> skew_pdiv((n+Sn)^3, Sn+n^2, n, A);
```

```
```
Noncommutative Euclidean Algorithm for $K<n,Sn>$

$\text{skew\_gcdex}(p,q,x,A)$

- The function $\text{skew\_gcdex}$ performs an extended skew gcd algorithm on the skew polynomials $p$ and $q$ viewed as polynomials in $x$ with coefficients in their other indeterminates. It returns a list $[g,a,b,u,v]$ such that $up+vq=0$ and $ap+bq=g$. Hence, $g$ is a gcd of $p$ and $q$ (in an algebra where all coefficient indeterminates are invertible), while $up$ and $vq$ are lcm's of $p$ and $q$.

```plaintext
> P := skew_product(S^n^2+n*Sn+3,Sn+n,A);
> Q := skew_product(S^n^3+n*Sn+3,Sn+n,A);
> G := skew_gcdex(P,Q,Sn,A);

This says the GCD of $(P,Q)$ is

```plaintext
> G[1];
```

```plaintext
9 n^2 + 54 Sn + 9 Sn n + 54 n
```

(Instead of working in $K(n)[Sn]$, we use only polynomials, i.e., $K<n,Sn>$, so this $K(n)$-multiple of what we expected $(Sn+n)$ was produced to keep denominators clear.)

This GCD can be expressed as a linear combination $g=a*P+b*Q$:

```plaintext
> skew_product(G[2],P,A)+skew_product(G[3],Q,A);
```

```plaintext
(-21 + 10 n^2 + 2 n + 2 n^3) Sn^4 + 54 n + (7 n^3 - 12 n + 9 n^2 + 5 n^4 + 18 + n^5) Sn
+ ((-15 + 9 n^3 + 5 n^2 + 2 n^4 - 8 n) Sn^2 + (12 n^2 - 19 n + 8 n^3 - 30 + n^4) Sn^3
+ ((-3 + 2 n + n^2) Sn^5 + (21 - 10 n^2 - 2 n - 2 n^3) Sn^4 + 9 n^2
+ ((36 - 7 n^3 - 9 n^2 - n^5 + 21 n - 5 n^4) Sn + (-9 n^3 + 15 - 5 n^2 - 2 n^4 + 8 n) Sn^2
+ ((-12 n^2 + 19 n - 8 n^3 + 30 - n^4) Sn^3 + (3 - 2 n - n^2) Sn^5
```

> collect("",Sn,factor);
```
> simplify("-G[1]"y; 

0 

Also, the LCM can be computed as a multiple of \( P \) or of \( Q \):

\[
\text{skew_product}(G[4], P, A) :: 
\begin{align*}
&\text{collect ("Sn", factor)}; \\
&(-n - 6) Sn^6 + (-2 n^2 - 47 - 20 n) Sn^5 + (-101 - n^3 - 14 n^2 - 64 n) Sn^4 \\
&\hspace{1cm} + (-134 - 24 n^2 - 95 n - 2 n^3) Sn^3 + (-n^4 - 141 n - 121 - 55 n^2 - 12 n^3) Sn^2 \\
&\hspace{1cm} + (-6 n^3 - 108 n - 54 n^2 - 114) Sn - 9 n (7 + n) \\
\end{align*}
\]

\[
\text{skew_product}(G[5], Q, A) :: 
\begin{align*}
&\text{collect ("Sn", factor)}; \\
&(6 + n) Sn^6 + (2 n^2 + 47 + 20 n) Sn^5 + (101 + n^3 + 14 n^2 + 64 n) Sn^4 \\
&\hspace{1cm} + (134 + 24 n^2 + 95 n + 2 n^3) Sn^3 + (n^4 + 141 n + 121 + 55 n^2 + 12 n^3) Sn^2 \\
&\hspace{1cm} + (6 n^3 + 108 n + 54 n^2 + 114) Sn + 9 n (7 + n) \\
\end{align*}
\]