Jordan Form and Functions of Matrices Examples

```
> restart;
  with(LinearAlgebra):
```

Example: Let's find the Jordan form J of the matrix A (defined below) and then find a "transition" matrix P such that $P^{-1}AP = J$.

```
> A := <<9,35,-1,16,46,-35,1>|<-5,-3,1,-8,-26,27,-5>|<-2,-22,2,-14,-18,2,
  -2>|<1,-5,-1,2,8,-15,1>|<7,21,1,18,36,-21,7>|<4,16,0,12,24,-12,4>|<-6,-30,
  2,-14,-54,50,2>>;
```

$$A := \begin{bmatrix} 9 & -5 & -2 & 1 & 7 & 4 & -6 \\ 35 & -3 & -22 & -5 & 21 & 16 & -30 \\ -1 & 1 & 2 & -1 & 1 & 0 & 2 \\ 16 & -8 & -14 & 2 & 18 & 12 & -14 \\ 46 & -26 & -18 & 8 & 36 & 24 & -54 \\ -35 & 27 & 2 & -15 & -21 & -12 & 50 \\ 1 & -5 & -2 & 1 & 7 & 4 & 2 \end{bmatrix} \quad (1)$$

First, I'll compute this using the JordanForm command:

```
> P := JordanForm(A,output='Q');

  J := P^(-1).A.P;
```

$$P := \begin{bmatrix} -4 & \frac{3}{4} & 64 & \frac{293}{151} & \frac{4941}{604} & -\frac{613}{151} & \frac{2395}{302} \\ 8 & -\frac{1}{2} & 192 & \frac{4627}{302} & \frac{56889}{2416} & -\frac{4131}{302} & \frac{55681}{2416} \\ 0 & 0 & 64 & -\frac{1679}{302} & \frac{20799}{2416} & -\frac{1377}{302} & \frac{20799}{2416} \\ 0 & -1 & 192 & \frac{1030}{151} & \frac{14803}{604} & -\frac{1990}{151} & \frac{14199}{604} \\ -4 & \frac{3}{4} & 64 & \frac{12817}{302} & \frac{10725}{2416} & -\frac{1377}{302} & \frac{12537}{2416} \\ 12 & -\frac{13}{4} & 128 & -\frac{6511}{151} & \frac{13513}{604} & -\frac{1377}{151} & \frac{5775}{302} \\ -4 & -\frac{1}{4} & 64 & \frac{293}{151} & \frac{4941}{604} & -\frac{613}{151} & \frac{2395}{302} \end{bmatrix}$$

(2)

$$J := \begin{bmatrix} 8 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} \tag{2}$$

Now let's build this ourselves. To begin with, let's find the Jordan form. Our (factored) characteristic polynomial is:

```
> factor(Determinant(t*IdentityMatrix(7)-A));
```

$$(t-8)^2 (t-4)^5 \tag{3}$$

Focus on the eigenvalue 8:

```
> n1 := 7-Rank((A-8*IdentityMatrix(7)));
  n2 := 7-Rank((A-8*IdentityMatrix(7))^2);
  n3 := 7-Rank((A-8*IdentityMatrix(7))^3);
```

$$n1 := 1$$
$$n2 := 2$$
$$n3 := 2 \tag{4}$$

Thus n1=1 tells us that there is only one linearly independent eigenvector with eigenvalue 8. Thus we only have 1 chain. Since the algebraic multiplicity of 8 is 2 and n2=n3=2, we have 2-1=1 chain of length 2. We need to find a vector solving $(A-8\,I)^2 x = 0$ but $(A-8\,I)x \neq 0$ (i.e., a generalized eigenvector of degree 2 - this will generate our chain).

```
> ReducedRowEchelonForm((A-8*IdentityMatrix(7))^2);
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \frac{1}{4} & -\frac{1}{4} \\ 0 & 1 & 0 & 0 & 0 & -\frac{1}{4} & \frac{5}{4} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -\frac{1}{4} & -\frac{3}{4} \\ 0 & 0 & 0 & 0 & 1 & \frac{1}{4} & -\frac{1}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

We have 2 free variables (i.e., K_8 = ker((A-8I)^2) is 2-dimensional). We have a basis for this kernel:
[If our free variables are s and t, I used s=4, t=0 and s=0, t=4 to clear fractions.]

```
> v := <<-1,1,0,1,-1,4,0>>;
  w := <<1,-5,0,3,1,0,4>>;
```

$$v := \begin{bmatrix} -1 \\ 1 \\ 0 \\ 1 \\ -1 \\ 4 \\ 0 \end{bmatrix}$$

$$w := \begin{bmatrix} 1 \\ -5 \\ 0 \\ 3 \\ 1 \\ 0 \\ 4 \end{bmatrix} \tag{6}$$

Either vector will do since both (A-8I)v and (A-8I)w are nonzero. We'll use v and define v1 = (A-8I)v and v2 = v our chain.

```
> v1 := (A-8*IdentityMatrix(7)).v;
  v2 := v;
```

$$v1 := \begin{bmatrix} 4 \\ -8 \\ 0 \\ 0 \\ 4 \\ -12 \\ 4 \end{bmatrix}$$

$$v2 := \begin{bmatrix} -1 \\ 1 \\ 0 \\ 1 \\ -1 \\ 4 \\ 0 \end{bmatrix} \tag{7}$$

Now we need to tackle the other eigenvalue (i.e., 4). Again, we compute some nullities to figure out our Jordan block structure.

```
> n1 := 7-Rank((A-4*IdentityMatrix(7)));
  n2 := 7-Rank((A-4*IdentityMatrix(7))^2);
  n3 := 7-Rank((A-4*IdentityMatrix(7))^3);
  n4 := 7-Rank((A-4*IdentityMatrix(7))^4);
```

$$n1 := 2$$
$$n2 := 4$$
$$n3 := 5$$
$$n4 := 5 \tag{8}$$

Since the nullity of A-4I is 2, we have 2 linearly independent eigenvectors (with eigenvalue 4). Thus there will be two chains. Notice that n2 - n1 = 4 - 2 = 2 so both chains are of length at least 2. Next, n3 - n2 = 5 - 4 = 1 so only one chain has length at least 3. Since n3 = n4 (so that n4 - n3 = 5 - 5 = 0), we have no longer chains. Therefore, we have a length 2 and a length 3 chain.

We find the longest chain(s) first. To that end we need a vector solving $(A - 4\,I)^3 x = 0$ but $(A - 4\,I)^2 x \neq 0$.

```
> ReducedRowEchelonForm((A-4*IdentityMatrix(7))^3);
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & -\frac{2}{3} & -1 & -\frac{1}{3} & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{9}$$

We have 5 free variables. I'll create a basis for ker((A-4I)^3) choosing values for these free variables that clear fractions (the one-thirds) and put these vectors together into a matrix:

```
> kBasis := <<0,2,3,0,0,0,0>|<0,1,0,1,0,0,0>|<0,1,0,0,3,0,0>|<0,0,0,0,0,1,
  0>|<1,-1,0,0,0,0,1>>;
```

$$kBasis := \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 2 & 1 & 1 & 0 & -1 \\ 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

Looking at (A-4*I)^2 times any one of these basis vectors...

```
> ((A-4*IdentityMatrix(7))^2).kBasis;
```

(11)

$$\begin{bmatrix} -136 & -24 & 88 & 32 & 24 \\ -408 & -72 & 264 & 96 & 72 \\ -136 & -24 & 88 & 32 & 24 \\ -408 & -72 & 264 & 96 & 72 \\ -136 & -24 & 88 & 32 & 24 \\ -272 & -48 & 176 & 64 & 48 \\ -136 & -24 & 88 & 32 & 24 \end{bmatrix} \tag{11}$$

...we see that all of the solutions yield $(A - 4\,I)^2 x \neq 0$. Thus we can choose the simplest choice (the fourth vector), call it v. Then we have a chain ((A-4I)^2)v, (A-4I)v, and v. Call these vectors w1, w2, and w3.

```
> v := <<0,0,0,0,0,1,0>>;

  w1 := ((A-4*IdentityMatrix(7))^2).v;
  w2 := (A-4*IdentityMatrix(7)).v;
  w3 := v;
```

$$v := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$w1 := \begin{bmatrix} 32 \\ 96 \\ 32 \\ 96 \\ 32 \\ 64 \\ 32 \end{bmatrix}$$

$$w2 := \begin{bmatrix} 4 \\ 16 \\ 0 \\ 12 \\ 24 \\ -16 \\ 4 \end{bmatrix}$$

$$w3 := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{12}$$

Finally, we need to find our next smallest (and last) chain. This has length 2. Thus we need to find a solution of $(A - 4\,I)^2 x = 0$ but we need x to be independent from everything in ker(A-4I) plus the span of w1, w2, and w3. We find bases for ker(A-4I) and then for ker((A-4I)^2):

```
> ReducedRowEchelonForm(A-4*IdentityMatrix(7));
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & -\frac{3}{2} & 0 \\ 0 & 0 & 1 & 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{13}$$

We have 2 free variables and again make choices to clear denominators and chuck these into basis vectors, for ker((A-4I)^2), into a matrix:

```
> k1Basis := <<0,3,1,2,1,2,0>|<1,0,0,1,0,0,1>>;
```

$$k1Basis := \begin{bmatrix} 0 & 1 \\ 3 & 0 \\ 1 & 0 \\ 2 & 1 \\ 1 & 0 \\ 2 & 0 \\ 0 & 1 \end{bmatrix} \tag{14}$$

Now for ker((A-4I)^2):

```
> ReducedRowEchelonForm((A-4*IdentityMatrix(7))^2);
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & -\frac{11}{17} & -\frac{13}{17} & -\frac{8}{17} & \frac{11}{17} \\ 0 & 0 & 1 & \frac{9}{17} & -\frac{11}{17} & -\frac{12}{17} & -\frac{9}{17} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{15}$$

We have 4 free variables and again chuck these into basis vectors, for ker((A-4I)^2), into a matrix:

```
> k2Basis := <<0,11,-9,17,0,0,0>|<0,13,11,0,17,0,0>|<0,8,12,0,0,17,0>|<17,
  -11,9,0,0,0,17>>;
```

$$k2Basis := \begin{bmatrix} 0 & 0 & 0 & 17 \\ 11 & 13 & 8 & -11 \\ -9 & 11 & 12 & 9 \\ 17 & 0 & 0 & 0 \\ 0 & 17 & 0 & 0 \\ 0 & 0 & 17 & 0 \\ 0 & 0 & 0 & 17 \end{bmatrix} \tag{16}$$

We need a solution that is independent from w1, w2, and w2 and our k1Basis. Thus we can just chuck w's and our k-Basis matrices together and row reduce to detect dependence:

```
> ReducedRowEchelonForm(<w1|w2|w3|k1Basis|k2Basis>);
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \frac{1}{32} & 0 & -\frac{17}{256} & \frac{17}{512} & \frac{17}{32} \\ 0 & 1 & 0 & 0 & 0 & 0 & \frac{17}{32} & -\frac{17}{64} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & \frac{51}{8} & \frac{85}{16} & -17 \\ 0 & 0 & 0 & 0 & 0 & 1 & -\frac{3}{4} & -\frac{5}{8} & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{17}$$

It turns out that all of our basis vectors were (individually) independent from w1, w2, w3, and the k1Basis vectors. Let's just pick the first one. Again call it v and then form our chain: u1 = (A-4I)v, u2 = v.

```
> v := <<0,11,-9,17,0,0,0>>;
```

```
u1 := (A-4*IdentityMatrix(7)).v;
u2 := v;
```

$$v := \begin{bmatrix} 0 \\ 11 \\ -9 \\ 17 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$u1 := \begin{bmatrix} -20 \\ 36 \\ 12 \\ 4 \\ 12 \\ 24 \\ -20 \end{bmatrix}$$

$$u2 := \begin{bmatrix} 0 \\ 11 \\ -9 \\ 17 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \textbf{(18)}$$

Assembling all of this together, we get our "transition" matrix P and thus our Jordan form as well:

```
> P := <v1|v2|w1|w2|w3|u1|u2>;

  J := P^(-1).A.P;
```

$$P := \begin{bmatrix} 4 & -1 & 32 & 4 & 0 & -20 & 0 \\ -8 & 1 & 96 & 16 & 0 & 36 & 11 \\ 0 & 0 & 32 & 0 & 0 & 12 & -9 \\ 0 & 1 & 96 & 12 & 0 & 4 & 17 \\ 4 & -1 & 32 & 24 & 0 & 12 & 0 \\ -12 & 4 & 64 & -16 & 1 & 24 & 0 \\ 4 & 0 & 32 & 4 & 0 & -20 & 0 \end{bmatrix}$$

(19)

$$J := \begin{bmatrix} 8 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} \tag{19}$$

Example: Let's find the Jordan form J of the matrix B (defined below) and then find a "transition" matrix P such that $P^{-1}BP=J$.

```
> B := <<2,-1,-1,-1>|<1,4,1,1>|<0,0,2,-1>|<0,0,1,4>>;
```

$$B := \begin{bmatrix} 2 & 1 & 0 & 0 \\ -1 & 4 & 0 & 0 \\ -1 & 1 & 2 & 1 \\ -1 & 1 & -1 & 4 \end{bmatrix} \tag{20}$$

First, we determine our eigenvalues and Jordan form structure:

```
> factor(Determinant(t*IdentityMatrix(4)-B));
```

$$(t-3)^4 \tag{21}$$

```
> n1 := 4-Rank(B-3*IdentityMatrix(4));
  n2 := 4-Rank((B-3*IdentityMatrix(4))^2);
  n3 := 4-Rank((B-3*IdentityMatrix(4))^3);
```

$$n1 := 2$$

$$n2 := 4$$

$$n3 := 4 \tag{22}$$

We only have one eigevalues (i.e., 3). n1=2 tells us that we'll have 2 chains. n2 - n1 = 4 - 2 = 2 says both chains have length at least 2. n3 - n2 = 4 - 4 = 0 says there are no chains of length 3 or longer. Thus we have 2 chains of length 2. Let's find our first chain. For a chain of length 2, we need to solve $(B-3\,I)^2\,x=0$ but $(B-3\,I)x\neq 0$. However, $(B-3\,I)^2$ is just the zero matrix (as expected - why?), so we can use the columns of the 4x4 identity as a basis for ker ((B-3I)^2). We just need to pick a vector such that $(B-3\,I)x\neq 0$. Notice that the first column of the identity matrix will do:

```
> v := <<1,0,0,0>>;

  (B-3*IdentityMatrix(4)).v;
```

$$v := \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

(23)

$$\begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad (23)$$

We get our first chain. Define v1 = (B-3I)v and v2 = v

```
> v1 := (B-3*IdentityMatrix(4)).v;
  v2 := v;
```

$$v1 := \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$$v2 := \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (24)$$

We need a second chain of length 2. Again, we need a vector that solves $(B-3\,I)^2 x = 0$ but is also indepedent from v1 and v2 and a basis for ker(B-3I). So to start let's find a basis for ker(B-3I):

```
> ReducedRowEchelonForm(B-3*IdentityMatrix(4));
```

$$\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (25)$$

```
> k1Basis := <<1,1,0,0>|<0,0,1,1>>;
```

$$k1Basis := \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad (26)$$

Since ker((B-3I)^2) is the whole space, we can use the columns of the identity matrix for its basis. Then we chuck v1, v2, and the k1Basis vectors into a matrix with k2Basis at the end to detect linear dependencies:

```
> k2Basis := IdentityMatrix(4);

  ReducedRowEchelonForm(<v1|v2|k1Basis|k2Basis>);
```

$$k2Basis := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{27}$$

It looks like the third and fourth vectors in k2Basis are independent from v1, v2, and k1Basis. Let's choose the third column for our v and then create our last chain: w1 = (B-3I)v and w2 = v.

```
> v := <<0,0,1,0>>;

  w1 := (B-3*IdentityMatrix(4)).v;
  w2 := v;
```

$$v := \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$w1 := \begin{bmatrix} 0 \\ 0 \\ -1 \\ -1 \end{bmatrix}$$

$$w2 := \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{28}$$

We assemble our findings:

```
> P := <v1|v2|w1|w2>;

  J := P^(-1).B.P;
```

$$P := \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -1 & 0 & -1 & 1 \\ -1 & 0 & -1 & 0 \end{bmatrix}$$

(29)

$$J := \begin{bmatrix} 3 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 3 \end{bmatrix} \quad (29)$$

Of course, the built in commands are a better choice (this gives a different transition matrix -- I like mine better):

```
> P := JordanForm(B,output='Q');
  J := P^(-1).B.P;
```

$$P := \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -2 & 1 & -1 & 1 \\ -2 & 0 & -1 & 0 \end{bmatrix}$$

$$J := \begin{bmatrix} 3 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 3 \end{bmatrix} \quad (30)$$

Example: Compute e^{C} and $\sin(C)$ for the matrix C defined below.

```
> C := <<1,0,1>|<2,2,-2>|<-1,0,3>>;
```

$$C := \begin{bmatrix} 1 & 2 & -1 \\ 0 & 2 & 0 \\ 1 & -2 & 3 \end{bmatrix} \quad (31)$$

First, we'll compute the Jordan form and transition matrix. I'll be more brief in my explaination:

```
> factor(Determinant(t*IdentityMatrix(3)-C));

  n1 := 3-Rank(C-2*IdentityMatrix(3));
  n2 := 3-Rank((C-2*IdentityMatrix(3))^2);
  n3 := 3-Rank((C-2*IdentityMatrix(3))^3);
```

$$(t-2)^3$$
$$n1 := 2$$
$$n2 := 3$$
$$n3 := 3 \quad (32)$$

We have two chains (one of length 1 and one of length 2). I'll find bases for ker(C-2I) and ker((C-2I)^2). Note that ker((C-2I)^2)=ker(0)=R^3, so we use the columns of the identity matrix for its basis.

```
> ReducedRowEchelonForm(C-2*IdentityMatrix(3));
```

(33)

$$\begin{bmatrix} 1 & -2 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (33)$$

We get our bases for these kernels and then choose a vector in ker((C-2I)^2) such that it is not in ker(C-2I) (the first column of the identity works). Then create our chain v1, v2.

```
> k1Basis := <<2,1,0>|<-1,0,1>>;
  k2Basis := IdentityMatrix(3);
  v := <<1,0,0>>;
  v1 := (C-2*IdentityMatrix(3)).v;
  v2 := v;
```

$$k1Basis := \begin{bmatrix} 2 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$k2Basis := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$v := \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$v1 := \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

$$v2 := \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (34)$$

Our second chain is just an eigenvector that's independent from v1 and v2:

```
> ReducedRowEchelonForm(<v1|v2|k1Basis>);
```

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (35)$$

The first basis vector in k1Basis works. (the second would not!)

```
> w1 := <<2,1,0>>;
```

$$w1 := \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix} \tag{36}$$

Assemble and finish Jordan form:

```
> P := <v1|v2|w1>;
  J := P^(-1).C.P;
```

$$P := \begin{bmatrix} -1 & 1 & 2 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$J := \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{37}$$

To compute functions of matrices we need the diagonal and nilpotent parts of our Jordan form stripped apart:

[Note: In Maple, "D" is a reserved letter, so we can't call the diagonal part D.]

```
> Diag := <<2,0,0>|<0,2,0>|<0,0,2>>;
  N := <<0,0,0>|<1,0,0>|<0,0,0>>;

  'J' = Diag + N;
```

$$Diag := \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$$N := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$J = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{38}$$

Exponentiating the diagonal part just mean exponentiating the diagonals. Exponentiating the nilpontent parts means computing: I + N + N^2/2! + N^3/3! + ... but N^2=N^3=...=0 so exp(N) is just I+N. Putting this together: exp(C) = P exp(D) exp(N) P^(-1).

```
> expD := <<exp(2),0,0>|<0,exp(2),0>|<0,0,exp(2)>>;
  expN := IdentityMatrix(3) + N;

  expC := P.expD.expN.P^(-1);
```

$$expD := \begin{bmatrix} e^2 & 0 & 0 \\ 0 & e^2 & 0 \\ 0 & 0 & e^2 \end{bmatrix}$$

$$expN := \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$expC := \begin{bmatrix} 0 & 2\,e^2 & -e^2 \\ 0 & e^2 & 0 \\ e^2 & -2\,e^2 & 2\,e^2 \end{bmatrix} \qquad \textbf{(39)}$$

Or the built in command:

```
> MatrixExponential(C);
```

$$\begin{bmatrix} 0 & 2\,e^2 & -e^2 \\ 0 & e^2 & 0 \\ e^2 & -2\,e^2 & 2\,e^2 \end{bmatrix} \qquad \textbf{(40)}$$

To compute the sine of a matrix, we need to break the diagonal and nilpotent computations apart. But we have the following identitiy: sin(x+y) = sin(x)cos(y)+cos(x)sin(y). By some deep algebraic magic, since this identity holds for real numbers, it also holds for matrices (I can explain if you'd like). Anyway, we can compute sin(J) = sin(D)cos(N)+ cos(D)sin(N). For sin(D) and cos(D), we simply apply those functions to the diagonals. To compute cos(N) and sin (N), note that cos(N) = I - N^2/2 + N^4/4! - ... = I and sin(N) = N - N^3/3! + N^5/5! - ... = N since N^2=0.

```
> sinD := <<sin(2),0,0>|<0,sin(2),0>|<0,0,sin(2)>>;
  cosD := <<cos(2),0,0>|<0,cos(2),0>|<0,0,cos(2)>>;

  sinN := N;
  cosN := IdentityMatrix(3);

  sinJ := sinD.cosN+cosD.sinN;

  sinC := P.sinJ.P^(-1);
```

$$sinD := \begin{bmatrix} \sin(2) & 0 & 0 \\ 0 & \sin(2) & 0 \\ 0 & 0 & \sin(2) \end{bmatrix}$$

$$cosD := \begin{bmatrix} \cos(2) & 0 & 0 \\ 0 & \cos(2) & 0 \\ 0 & 0 & \cos(2) \end{bmatrix}$$

$$sinN := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$cosN := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$sinJ := \begin{bmatrix} \sin(2) & \cos(2) & 0 \\ 0 & \sin(2) & 0 \\ 0 & 0 & \sin(2) \end{bmatrix}$$

$$sinC := \begin{bmatrix} -\cos(2) + \sin(2) & 2\cos(2) & -\cos(2) \\ 0 & \sin(2) & 0 \\ \cos(2) & -2\cos(2) & \sin(2) + \cos(2) \end{bmatrix} \quad \textbf{(41)}$$

Is this right? Well, if we approximate sin(C) by adding up the first bunch of terms in its series and then use a decimal approximation, we can see that this matches what we found:

```
> seriesApprox = evalf(sum((-1)^k*C^(2*k+1)/(2*k+1)!,k=0..10));
  sinCApprox = evalf(sinC);
```

$$seriesApprox = \begin{bmatrix} 1.325444263 & -0.8322936731 & 0.4161468365 \\ 0. & 0.9092974268 & 0. \\ -0.4161468365 & 0.8322936731 & 0.4931505903 \end{bmatrix}$$

$$sinCApprox = \begin{bmatrix} 1.325444263 & -0.8322936730 & 0.4161468365 \\ 0. & 0.9092974268 & 0. \\ -0.4161468365 & 0.8322936730 & 0.4931505903 \end{bmatrix} \quad \textbf{(42)}$$

There is a built in function to compute functions of matrices:

```
> MatrixFunction(C,sin(x),x);
```

$$\begin{bmatrix} -\cos(2) + \sin(2) & 2\cos(2) & -\cos(2) \\ 0 & \sin(2) & 0 \\ \cos(2) & -2\cos(2) & \sin(2) + \cos(2) \end{bmatrix} \quad \textbf{(43)}$$

Computing the square root of non-diagonalizable matrices is pretty tricky (they don't always exist - even over the complex numbers). The built in function will do it for us:

```
> sqrtC := MatrixFunction(C,sqrt(x),x);
```

$$sqrtC := \begin{bmatrix} \frac{3\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{4} \\ 0 & \sqrt{2} & 0 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{5\sqrt{2}}{4} \end{bmatrix} \quad \textbf{(44)}$$

```
> 'sqrt(C)^2' = sqrtC^2;
```

(45)

$$\left(\sqrt{C}\right)^2 = \begin{bmatrix} 1 & 2 & -1 \\ 0 & 2 & 0 \\ 1 & -2 & 3 \end{bmatrix} \tag{45}$$

Example: Compute $f(M)$ where $f(x)$ is a function defined by a series (and converges wherever it needs to) and M is the matrix defined below.

```
> M := <<-11,-12,12>|<12,13,-12>|<4,4,-3>>;
```

$$M := \begin{bmatrix} -11 & 12 & 4 \\ -12 & 13 & 4 \\ 12 & -12 & -3 \end{bmatrix} \tag{46}$$

Compute the Jordan form:

```
> factor(Determinant(t*IdentityMatrix(3)-M));
```

$$(t+3)\,(t-1)^2 \tag{47}$$

```
> ReducedRowEchelonForm(M-(-3)*IdentityMatrix(3));
```

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{48}$$

```
> v1 := <<-1,-1,1>>;
```

$$v1 := \begin{bmatrix} -1 \\ -1 \\ 1 \end{bmatrix} \tag{49}$$

```
> n1 := 3-Rank(M-1*IdentityMatrix(3));
  n2 := 3-Rank((M-1*IdentityMatrix(3))^2);
```

$$n1 := 2$$

$$n2 := 2 \tag{50}$$

```
> ReducedRowEchelonForm(M-1*IdentityMatrix(3));
```

$$\begin{bmatrix} 1 & -1 & -\frac{1}{3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{51}$$

```
> w1 := <<1,1,0>>;
  u1 := <<1,0,3>>;
```

$$w1 := \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

$$u1 := \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix} \tag{52}$$

```
> P := <v1|w1|u1>;
```

```
J := P^(-1).M.P;
```

$$P := \begin{bmatrix} -1 & 1 & 1 \\ -1 & 1 & 0 \\ 1 & 0 & 3 \end{bmatrix}$$

$$J := \begin{bmatrix} -3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{53}$$

M is diagonalizable! This makes everything much much simpler.

```
> fJ := <<f(-3),0,0>|<0,f(1),0>|<0,0,f(1)>>;
  fM := P.fJ.P^(-1);
```

$$fJ := \begin{bmatrix} f(-3) & 0 & 0 \\ 0 & f(1) & 0 \\ 0 & 0 & f(1) \end{bmatrix}$$

$$fM := \begin{bmatrix} 3 f(-3) - 2 f(1) & -3 f(-3) + 3 f(1) & -f(-3) + f(1) \\ 3 f(-3) - 3 f(1) & -3 f(-3) + 4 f(1) & -f(-3) + f(1) \\ -3 f(-3) + 3 f(1) & 3 f(-3) - 3 f(1) & f(-3) \end{bmatrix} \tag{54}$$

What about specific functions?

Matrix Exponential?

```
> f := x -> exp(x):
  'f(x)' = f(x);
  expM := simplify(fM);
```

$$f(x) = e^x$$

$$expM := \begin{bmatrix} (-2 e^4 + 3) e^{-3} & 3 (e^4 - 1) e^{-3} & (e^4 - 1) e^{-3} \\ -3 (e^4 - 1) e^{-3} & 4 e^{-3} e^4 - 3 e^{-3} & (e^4 - 1) e^{-3} \\ 3 (e^4 - 1) e^{-3} & -3 (e^4 - 1) e^{-3} & e^{-3} \end{bmatrix} \tag{55}$$

```
> MatrixExponential(M);
```

$$\begin{bmatrix} (-2 e^4 + 3) e^{-3} & 3 (e^4 - 1) e^{-3} & (e^4 - 1) e^{-3} \\ -3 (e^4 - 1) e^{-3} & 4 e^{-3} e^4 - 3 e^{-3} & (e^4 - 1) e^{-3} \\ 3 (e^4 - 1) e^{-3} & -3 (e^4 - 1) e^{-3} & e^{-3} \end{bmatrix} \tag{56}$$

Sine?

```
> f := x -> sin(x):
  'f(x)' = f(x);
  sinM := simplify(fM);
```

$$f(x) = \sin(x)$$

$$sinM := \begin{bmatrix} -3\sin(3) - 2\sin(1) & 3\sin(3) + 3\sin(1) & \sin(3) + \sin(1) \\ -3\sin(3) - 3\sin(1) & 3\sin(3) + 4\sin(1) & \sin(3) + \sin(1) \\ 3\sin(3) + 3\sin(1) & -3\sin(3) - 3\sin(1) & -\sin(3) \end{bmatrix} \quad (57)$$

Square root?

```
> f := x -> sqrt(x):
  'f(x)' = f(x);

  sqrtM := simplify(fM);
```

$$f(x) = \sqrt{x}$$

$$sqrtM := \begin{bmatrix} 3\,\mathrm{I}\sqrt{3} - 2 & -3\,\mathrm{I}\sqrt{3} + 3 & -\mathrm{I}\sqrt{3} + 1 \\ 3\,\mathrm{I}\sqrt{3} - 3 & -3\,\mathrm{I}\sqrt{3} + 4 & -\mathrm{I}\sqrt{3} + 1 \\ -3\,\mathrm{I}\sqrt{3} + 3 & 3\,\mathrm{I}\sqrt{3} - 3 & \mathrm{I}\sqrt{3} \end{bmatrix} \quad (58)$$

```
> 'M' = simplify(sqrtM^2);
```

$$M = \begin{bmatrix} -11 & 12 & 4 \\ -12 & 13 & 4 \\ 12 & -12 & -3 \end{bmatrix} \quad (59)$$

Alternatively, we could use MatrixFunction:

```
> MatrixFunction(M,exp(x),x);
  MatrixFunction(M,sin(x),x);
  MatrixFunction(M,sqrt(x),x);
```

$$\begin{bmatrix} (-2\,e^4 + 3)\,e^{-3} & 3\,(e^4 - 1)\,e^{-3} & (e^4 - 1)\,e^{-3} \\ -3\,(e^4 - 1)\,e^{-3} & 4\,e^{-3}\,e^4 - 3\,e^{-3} & (e^4 - 1)\,e^{-3} \\ 3\,(e^4 - 1)\,e^{-3} & -3\,(e^4 - 1)\,e^{-3} & e^{-3} \end{bmatrix}$$

$$\begin{bmatrix} -3\sin(3) - 2\sin(1) & 3\sin(3) + 3\sin(1) & \sin(3) + \sin(1) \\ -3\sin(3) - 3\sin(1) & 3\sin(3) + 4\sin(1) & \sin(3) + \sin(1) \\ 3\sin(3) + 3\sin(1) & -3\sin(3) - 3\sin(1) & -\sin(3) \end{bmatrix}$$

$$\begin{bmatrix} 3\,\mathrm{I}\sqrt{3} - 2 & -3\,\mathrm{I}\sqrt{3} + 3 & -\mathrm{I}\sqrt{3} + 1 \\ 3\,\mathrm{I}\sqrt{3} - 3 & -3\,\mathrm{I}\sqrt{3} + 4 & -\mathrm{I}\sqrt{3} + 1 \\ -3\,\mathrm{I}\sqrt{3} + 3 & 3\,\mathrm{I}\sqrt{3} - 3 & \mathrm{I}\sqrt{3} \end{bmatrix} \quad (60)$$