Dror Bar-Natan: Classes: 2014-15: Math 1100 Algebra I:

Running the JCF Programs

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\Classes\\14-1100"];
<< JCF-Program.m
```

Matrix 1 - 3x3, 3 eigenvalues.

```
n = 3; AA = {{3, 0, 0}, {4, -2, -6}, {-2, 0, 1}};
PP = QQ = II = IdentityMatrix[n];
MM = x II - AA; NN = PP.MM.QQ;

SwapRows[1, 2]

SwapColumns[2, 3]

GCDTrick[{1, 2}, 1]

GCDTrick[1, {1, 2}]

GCDTrick[1, {1, 3}]

GCDTrick[{1, 3}, 1]

GCDTrick[2, {2, 3}]

GCDTrick[{2, 3}, 2]

GCDTrick[2, {2, 3}]

SplitToSum[1, 3, (-3 + x), (-2 + x + x^2)]

Factor[-2 + x + x^2]

SplitToSum[2, 3, (-1 + x), (2 + x)]

MatrixForm /@ {Coefficient[NN, x, 1], BB = -Coefficient[NN, x, 0]}

(CC = Sum[
    MatrixPower[BB, k].Coefficient[PP, x, k],
    {k, 0, n}
   ]) // MatrixForm

CC.AA.Inverse[CC] // MatrixForm
```

Matrix 2 - 3x3, one Jordan block.

```
n = 3; AA = {{-5/2, -11, 9/2}, {-1/2, 1, 1/2}, {-19/2, -16, 21/2}};
```

```
PP = QQ = II = IdentityMatrix[n];
MM = x II - AA;  NN = PP.MM.QQ;
```

```
GCDTrick[1, {1, 2}]
```

```
GCDTrick[1, {1, 3}]
```

```
GCDTrick[{1, 2}, 1]
```

```
GCDTrick[{1, 3}, 1]
```

```
GCDTrick[2, {2, 3}]
```

```
GCDTrick[{2, 3}, 2]
```

```
JordanTrick[2, 3, x - 3, 3]
```

```
JordanTrick[1, 2, x - 3, 2]
```

```
MatrixForm /@ {Coefficient[NN, x, 1], BB = -Coefficient[NN, x, 0]}
```

```
(CC = Sum[
    MatrixPower[BB, k].Coefficient[PP, x, k],
    {k, 0, n}
   ]) // MatrixForm
```

```
CC.AA.Inverse[CC] // MatrixForm
```

Matrix 3 - 4x4, mixed Jordan form.

```
n = 4; AA = {{1, -2, 0, -2}, {1/4, 5/2, 0, 3/2}, {5/2, 5, 2, 3}, {1/4, 1/2, 0, 3/2}};
```

```
PP = QQ = II = IdentityMatrix[n];
MM = x II - AA;  NN = PP.MM.QQ;
```

```
GCDTrick[{1, 2}, 1]
```

```
GCDTrick[1, {1, 2}]
```

```
GCDTrick[1, {1, 4}]
```

```
GCDTrick[{1, 3}, 1]

GCDTrick[{1, 4}, 1]

GCDTrick[2, {2, 3}]

GCDTrick[2, {2, 4}]

GCDTrick[{2, 3}, 2]

GCDTrick[{2, 4}, 2]

GCDTrick[3, {3, 4}]

SwapBoth[1, 3]

SplitToSum[2, 4, (-1 + x), (-2 + x)^2]

SwapBoth[1, 2]

JordanTrick[3, 4, x - 2, 2]

MatrixForm /@ {Coefficient[NN, x, 1], BB = -Coefficient[NN, x, 0]}

(CC = Sum[
    MatrixPower[BB, k].Coefficient[PP, x, k],
    {k, 0, n}
   ]) // MatrixForm

CC.AA.Inverse[CC] // MatrixForm
```