```
> with(LinearAlgebra): with(plots):
```

Some comments about span and linear independence in Chapter 4...

Showing something does not span To show something does not span R^n, we must find a vector that is not in the span of the set. Setting up the homogeneous solution does not help for span. For example, infinitely many solutions to the homogeneous system says nothing about span [(1,0),(0,1),(1,1) is not li and has infinitely many solutions to the homogeneous equation, but it still spans R^2]- you need to find a vector that can't be represented (ie inconsistent system). Look at the four vectors below.

```
> M:=Matrix([[1,1,1,1],[2,2,2,2],[1,3,2,4]]);
```

$$M := \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 3 & 2 & 4 \end{bmatrix}$$

Let's set up the corresponding augmented system to test:

```
> H:=Matrix([[1,1,1,1,u1],[2,2,2,2,u2],[1,3,2,4,u3]]);
```

$$H := \begin{bmatrix} 1 & 1 & 1 & 1 & u1 \\ 2 & 2 & 2 & 2 & u2 \\ 1 & 3 & 2 & 4 & u3 \end{bmatrix}$$

```
> GaussianElimination(H);
```

$$\begin{bmatrix} 1 & 1 & 1 & 1 & u1 \\ 0 & 2 & 1 & 3 & u3-u1 \\ 0 & 0 & 0 & 0 & u2-2\,u1 \end{bmatrix}$$

We see that we won't always have a solution. If we pick u2 = 1 and u1 = 1 then the bottom row will give us 0 0 0 0 -1, ie no solutions. So we can take (1,1,0) for example.

Since (1,1,0) cannot be expressed as a combination of the other four vectors, the original 4 vectors do not span R^3. They do span a subspace in R^3. Since they are not all on the same line through the origin (they are not multiples of each other), they span a plan through the origin in R^3. There are other examples that are not l.i. but DO span - for example, {(1,0,0),(0,1,0),(0,0,1),(1,2,3)}. Since the first three are already a basis for R^3, they already span R^3, and adding one more vector destroys efficency (ie not li) but they will still span.

Visualization Here are the original 4 linear dependent vectors along with (1,1,0) sticking out of the plane. I've rotated the picture so that we see the plane "head-on."

```
> a1:=spacecurve({[1*t,2*t,1*t,t=0..1]},color=red, thickness=2):
   a2:=textplot3d([1,2,1,  `          vector [1,2,1]`],color=black):
```

```
b1:=spacecurve({[1*t,2*t,3*t,t=0..1]},color=green, thickness=2):
b2:=textplot3d([1,2,3,  `              vector [1,2,3]`],color=black):
c1:=spacecurve({[1*t,2*t,2*t,t=0..1]},color=magenta, thickness=2):
c2:=textplot3d([1,2,2,  `              vector [1,2,2]`],color=black):
d1:=spacecurve({[1*t,2*t,4*t,t=0..1]},color=black, thickness=2):
d2:=textplot3d([1,2,4,  `              vector [1,2,4]`],color=black):
e1:=spacecurve({[1*t,1*t,0*t,t=0..1]},color=yellow, thickness=2):
e2:=textplot3d([1,1,0,  `              vector [1,1,0]`],color=black):
display(a1,a2,b1,b2,c1,c2,d1,d2,e1,e2);
```
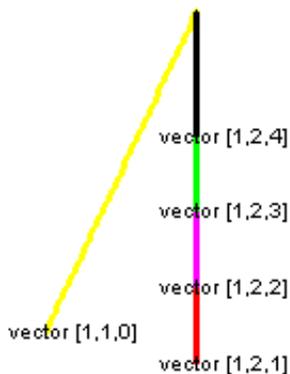

Testing for Linear Independence If I set up the following matrix containing three vetors below and reduce,

```
> M:=Matrix([[1,-1,1],[0,0,2],[2,-2,3]]);
```

$$M := \begin{bmatrix} 1 & -1 & 1 \\ 0 & 0 & 2 \\ 2 & -2 & 3 \end{bmatrix}$$

```
> ReducedRowEchelonForm(M);
```

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

we see that the third vector cannot be written as a linear combination of the first two. This DOES NOT imply that these three vectors are linearly independent, as one of the other vectors might be a combination of those that remain. We can not apply the shortcut theorem yet to say this forms a basis for R^3: Let's set up the homogeneous system formed by Mx=0:

```
> P:=Matrix([[1,-1,1,0],[0,0,2,0],[2,-2,3,0]]);
```

$$P := \begin{bmatrix} 1 & -1 & 1 & 0 \\ 0 & 0 & 2 & 0 \\ 2 & -2 & 3 & 0 \end{bmatrix}$$

```
> ReducedRowEchelonForm(P);
```

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

So we see that there are infinitely many solutions $c_3=0$, $c_2=t$, $c_1=t$ to the homogeneous system. Hence these vectors are linearly dependent, and do not form a basis.

Shortcut Theorem Also, the shortcut theorem for a basis only applies if you already have the correct number of vectors. Statements like "these are linearly dependent so they do not span" are false if I don't have the correct number of vectors already and apply the shortcut theorem. I can span but not be l.i. (represent but not be efficient) by having more vectors than I need. I can be l.i. but not span (be efficient but not represent eveything - only a subspace instead) by having less vectors than I need.