Third Tutorial

```
> with(plots):
> with(LinearAlgebra):
Warning, the name changecoords has been redefined
```

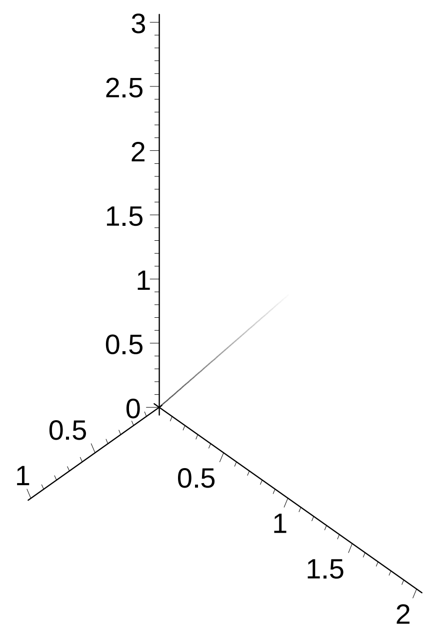
We want to visualize vectors in R^3. Of course, vectors can be thought of as points in R^3 and we can plot them as points. But it also helps if we can see the "arrow" going from the origin to the "tip" of the vector. We do that as follows.

Say that we want to visualize the vector v1=[1,2,3]. We use the command `spacecurve`:

```
> v1:=spacecurve([t,2*t,3*t],t=0..1):
```

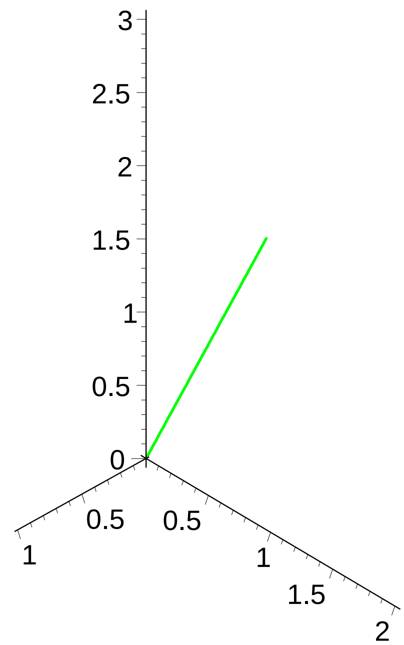
To actually see v1, we use `display`, adding the axes and scaling options to make sure that we see the axes and that the proportions are not distorted.

```
> display(v1,axes=normal,scaling=constrained);
```



The problem is that it is still hard to see v1. We make it green an thicker and try again:

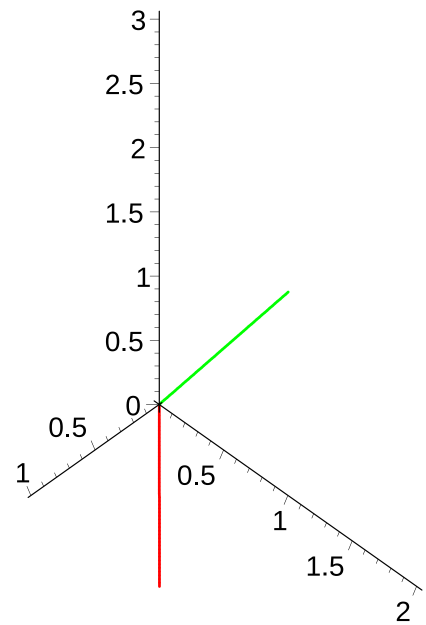
```
> v1:=spacecurve([t,2*t,3*t],t=0..1,color=green,thickness=3):
  display(v1,axes=normal,scaling=constrained);
```



Make sure that you rotate the previous image to get a feeling for what v1 looks like in three dimensions.

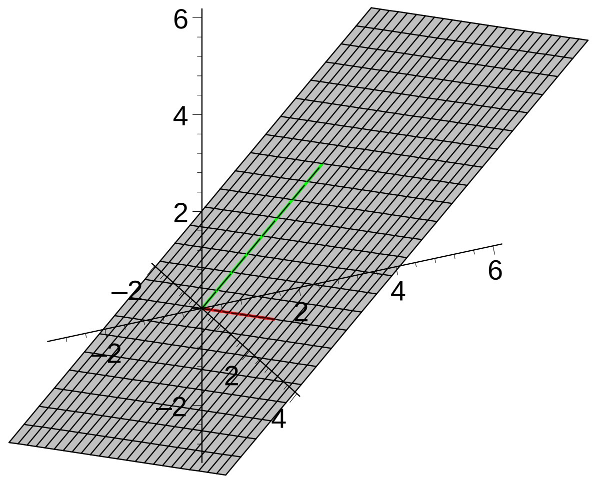
We want to add to the graph a second vector, v2=[1,1,0]:

```
> v2:=spacecurve([t,t,0],t=0..1,color=red,thickness=3):
  display(v1,v2,axes=normal,scaling=constrained);
```



Next we want to consider the span of v1 and v2, that is, the set of all vectors obtained as linear combinations of v1 and v2: a*v1+b*v2 for all possible a and b. Explicitly this is the set of vectors [a*1+b*1,a*2+b*1,c*3], that we plot using the `plot3d` command.

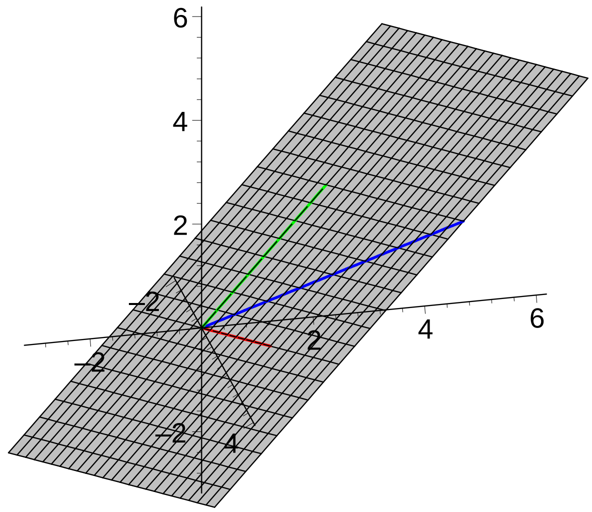
```
> p12:=plot3d([a+b,2*a+b,3*a],a=-1..2,b=-1..2,color=gray):
  display(p12,v1,v2,axes=normal,scaling=constrained);
```



Again, make sure that you rotate the previous graph to see the relation between the objects involved. As we know, the span of v1 and v2 is a subspace of dimension 2 in R^3 (that is, a plane), that we have called p12. Notice that {v1, v2} is a basis of p12.

Now, say that the vector w has coordinates [1,2] with respect to the basis {v1, v2}. We want to incorporate w to the previous graph. First remember that using the coordinates, w = 1*v1+2*v2 = 1*[1,2,3]+2*[1,1,0] = [3,4,3].

```
> w:=spacecurve([3*t,4*t,3*t],t=0..1,color=blue,thickness=3):
  display(p12,v1,v2,w,axes=normal,scaling=constrained);
```



Consider now w=[4,1,0]. Can we write w as a linear combination of v1 and v2 and, if so, what are the coefficients? One way of solving this problem is to write down exactly what it means to be a linear combination: w = a*v1 + b*v2, or, explicitly:
[4,1,0] = [a*1+b*1, a*2+b*1, a*3]
Finding a and b involves solving the linear system of equations whose augmented matrix is

```
> A:=<<1,2,3>|<1,1,0>|<4,1,0>>;
```

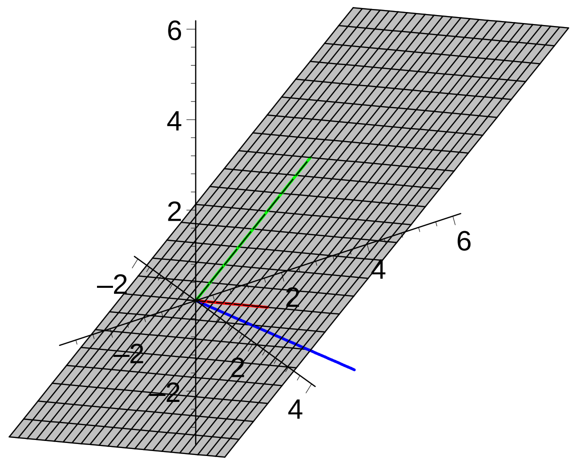
$$A := \begin{bmatrix} 1 & 1 & 4 \\ 2 & 1 & 1 \\ 3 & 0 & 0 \end{bmatrix}$$

We solve the problem with `LinearSolve`

```
> LinearSolve(A);
Error, (in LinearAlgebra:-LA_Main:-LinearSolve) inconsistent system
```

And we see that the system has no solution. This means that w is not a linear combination of v1 and v2.
Graphically, this result can be seen as follows: if w were a linear combination of v1 and v2, w should be in the plane p12. We can see if that is true or not:

```
> w:=spacecurve([4*t,1*t,0],t=0..1,color=blue,thickness=3):
  display(p12,v1,v2,w,axes=normal,scaling=constrained);
```



and we see that w is not in p12, so that it cannot be a linear combination of v1 and v2.

In many circumstances you may want to save some information to a file. For instance, above we have defined a Matrix A, as well as some graphics (spacecurves) v1 and v2. We can save them for later use to a file that we name "kk.m" as follows:

```
> save A, v1, v2, "kk.m";
```

We can clear A, that is, clear its value

```
> A;
  A:='A';
  A;
```

$$\begin{bmatrix} 1 & 1 & 4 \\ 2 & 1 & 1 \\ 3 & 0 & 0 \end{bmatrix}$$

$$A := A$$

$$A$$

To recover the value of A (as well as the other two variables) we read them from the file "kk.m"

```
> read "kk.m";
  A;
```

$$\begin{bmatrix} 1 & 1 & 4 \\ 2 & 1 & 1 \\ 3 & 0 & 0 \end{bmatrix}$$