

An Extended Example of triple integration

```
> with(plottools):  
with(plots):
```

This worksheet considers problem 15 from section 16.3. It is worthwhile to do an extended look at the example.

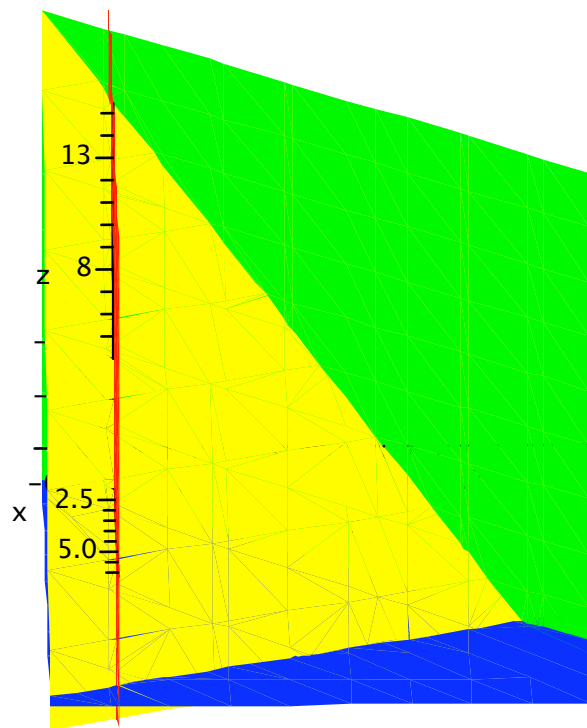
Visualizing the region

One of the big questions from the section is to be able to visualize the region of integration and set up the integral.

We do this with the `implicitplot3d` command, color coding the 4 planes.

```
> eqn1 := z=-6;  
eqn2 := y=0;  
eqn3 := y-x=4;  
eqn4 := 2*x+y+z=4;  
planes := implicitplot3d([eqn1, eqn2, eqn3, eqn4], x=-5..6,  
y=-1..7, z=-7..15,  
color=[blue, red, green, yellow], style=patchnogrid,  
transparency=.5, axes=normal):  
display(planes);
```

```
eqn1 := z = -6  
eqn2 := y = 0  
eqn3 := y - x = 4  
eqn4 := 2x + y + z = 4
```



A first activity is to drag the plot around so that we look at it down each axis in turn. We want to note which axes give a nice front and back to the tetrahedron.

If we look down the z-axis, the red and green planes become border lines, and the yellow and blue planes become front and back.

If we look down the x-axis, the red and blue planes become border lines, and the yellow and green planes become front and back.

If we look down the y-axis, the blue plane becomes a border line, the red plane becomes the back, and the yellow and green planes each become part of the front. This probably means we don't want the y-axis as the inside integration.

We would also like to find the coordinates of the corners of the tetrahedron and add them to the picture.

```
> p1 := solve({eqn2, eqn3, eqn4}, [x, y, z]);
point1 := <rhs(p1[1,1]), rhs(p1[1,2]), rhs(p1[1,3])>;
p2 := solve({eqn1, eqn3, eqn4}, [x, y, z]);
point2 := <rhs(p2[1,1]), rhs(p2[1,2]), rhs(p2[1,3])>;
p3 := solve({eqn1, eqn2, eqn4}, [x, y, z]);
point3 := <rhs(p3[1,1]), rhs(p3[1,2]), rhs(p3[1,3])>;
p4 := solve({eqn1, eqn2, eqn3}, [x, y, z]);
point4 := <rhs(p4[1,1]), rhs(p4[1,2]), rhs(p4[1,3])>;
points := pointplot3d([point1, point2, point3, point4],
    symbol=diamond, symbolsize=30, color=gold, axes=normal);
display([points, planes]);
```

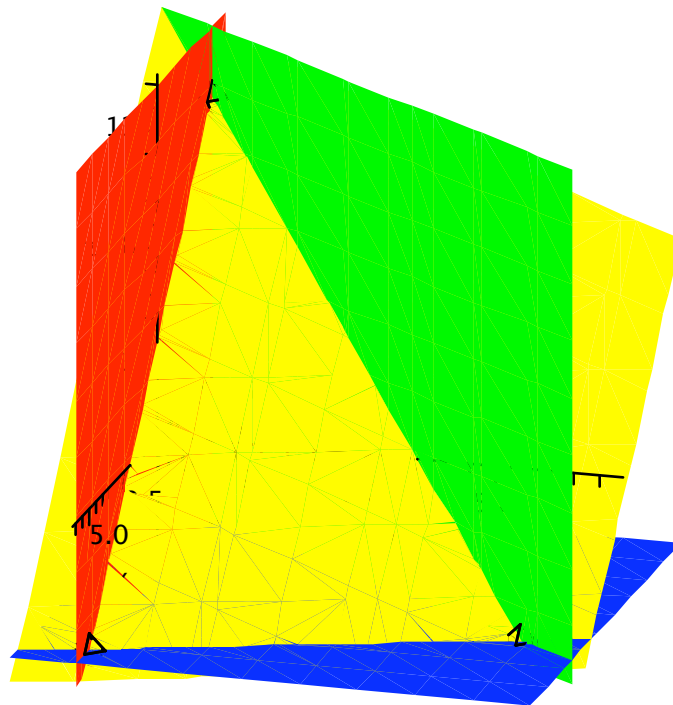
```
p1 := [[x = -4, y = 0, z = 12]]
```

```
point1 := [ -4  
            0  
            12 ]
```

```
point2 := [ 2  
            6  
            -6 ]
```

```
point3 := [ 5  
            0  
            -6 ]
```

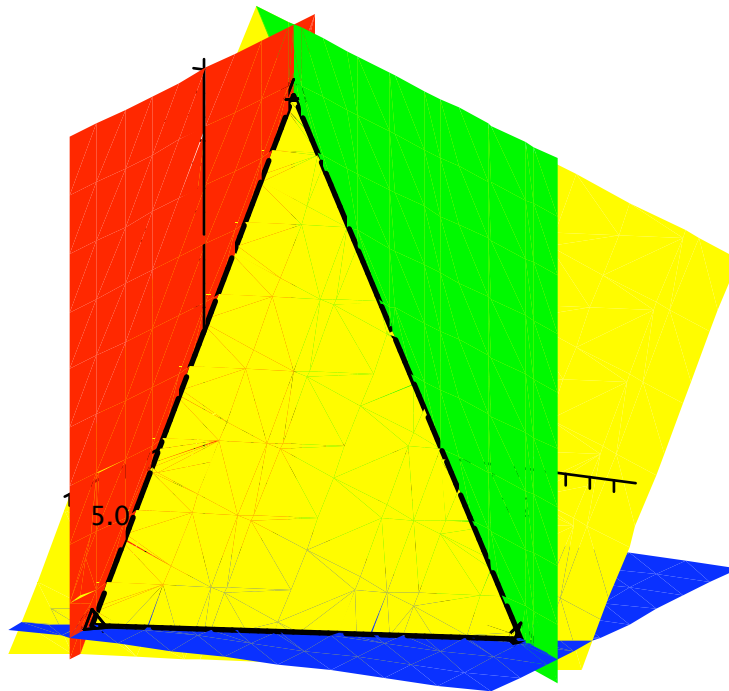
```
point4 := [ -4  
            0  
            -6 ]
```



With the vertices known, it is fairly easy to create the edges as parameterized curves using the trick

that $t*q1+(1-t)*q2$ goes from $q1$ to $q2$ as t goes from 0 to 1.

```
> line1 := t*point1 + (1-t)*point2:  
line2 := t*point1 + (1-t)*point3:  
line3 := t*point1 + (1-t)*point4:  
line4 := t*point2 + (1-t)*point3:  
line5 := t*point2 + (1-t)*point4:  
line6 := t*point3 + (1-t)*point4:  
lines := spacecurve({line1, line2, line3, line4, line5,  
line6},  
t=0..1,color=black, thickness=2):  
display([points,planes, lines]);
```



▼ Converting the visualization to a triple integral

We would now like to set up the integral. We will do this by first going from the inside out. For the first integral we would like an arrangement that goes from front to back with an easy description of the front and back. As noted above, we can then easily work with x or z as the inside variable. We will choose to work with x .

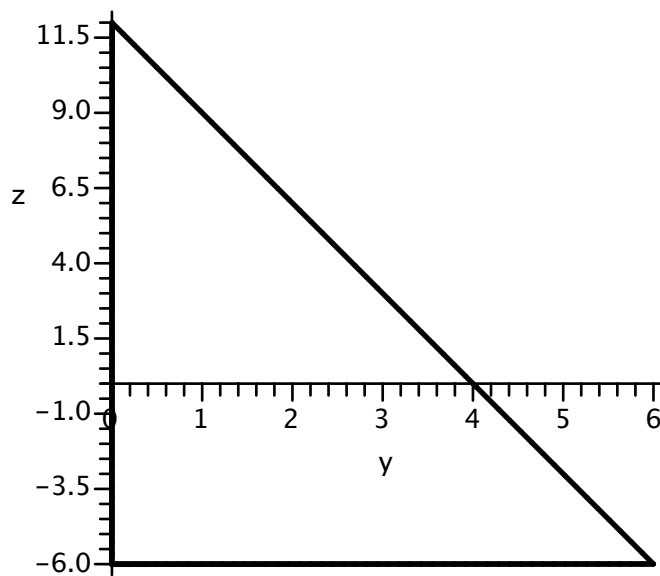
If we look down the x -axis with high values toward us, green is the back and yellow is the front. Solving the appropriate plane equations, we integrate from $x=y-4$ to $x=(4-y-z)/2$.

```
> I1 := Int(integrand, x=(y-4)..(4-y-z)/2);
```

$$I1 := \int_{y=4}^{2 - \frac{1}{2}y - \frac{1}{2}z} \text{integrand } dx \quad (2.1)$$

It is worthwhile to look at the frame of our tetrahedron again, from the viewpoint of looking down the x axis.

```
> display(lines, orientation=[0,90], axes=normal, labels=["x", "y", "z"]);
```



We see that we can next integrate in either y or z. If we integrate in y next we go from the line $y=0$ (the projection of the red plane on the y-z plane) to the line $y=4-z/3$ (the intersection of eq3 and eq4, the yellow and green planes, projected into the y-z plane).

```
> solve({eqn3,eqn4},{x,y});
```

$$\left\{ y = 4 - \frac{1}{3}z, x = -\frac{1}{3}z \right\} \quad (2.2)$$

```
> I2 := Int(I1, y=0..(4-z/3));
```

$$I2 := \int_0^{4 - \frac{1}{3}z} \int_{y=4}^{2 - \frac{1}{2}y - \frac{1}{2}z} \text{integrand } dx \, dy \quad (2.3)$$

That leaves us ready to integrate with respect to z. Looking at the vertices, we see that z goes from -6 to 12.

> **I3 := Int(I2, z=-6..12);**

$$I3 := \int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} \text{integrand} \, dx \, dy \, dz \quad (2.4)$$

To find the volume, we set integrand to 1.

> **I31 := eval(I3, integrand=1);**
value(I31);

$$I31 := \int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} 1 \, dx \, dy \, dz = 162 \quad (2.5)$$

To find the x coordinate of the center of gravity we divide the integral with integrand x by the integral with integrand 1.

> **cgx := subs(integrand=x, I3)/eval(I3, integrand=1);**
value(cgx);
cgy := subs(integrand=y, I3)/eval(I3, integrand=1);
value(cgy);
cgz := subs(integrand=z, I3)/eval(I3, integrand=1);
value(cgz);

$$cgx := \frac{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} x \, dx \, dy \, dz}{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} 1 \, dx \, dy \, dz} = \frac{1}{4}$$

$$cgy := \frac{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} y \, dx \, dy \, dz}{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} 1 \, dx \, dy \, dz} = \frac{3}{2}$$

$$cgz := \frac{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} z \, dx \, dy \, dz}{\int_{-6}^{12} \int_0^{4 - \frac{1}{3}z} \int_{y-4}^{2 - \frac{1}{2}y - \frac{1}{2}z} 1 \, dx \, dy \, dz} = \frac{3}{2} \quad (2.6)$$

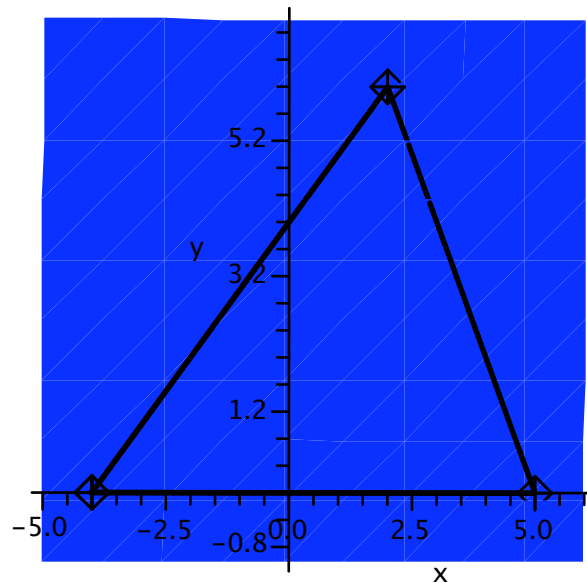
Parameterizing the surfaces and looking at slices

We would like to be able to replace the planes in the picture with the triangular regions. To do that we need to parameterize each region. We start with blue and red since they are easiest.

For the blue plane we want to look down the z-axis (orientation=[270,0]) where the triangular section becomes a simple triangle in the plane. The three vertices on the plane are (2,6), (5,0), and (-4,0).

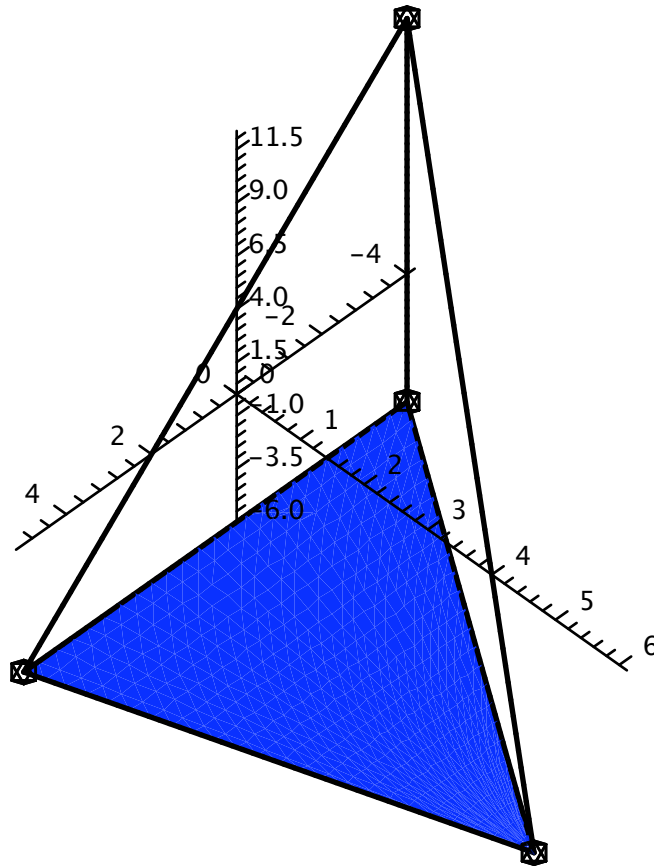
```
> eqn1;  
planeBlue := implicitplot3d([eqn1], x=-5..6, y=-1..7, z=-7.  
.15,  
color=[blue], style=patchnogrid,  
transparency=.5, axes=normal):  
display([points, lines, planeBlue], axes=normal, orientation=  
[270,0]);
```

$$z = -6$$



The z value is always -6. The y values go from 0 to 6. For each y value x goes from y-4 to -y/2+5. We parameterize x with the same $q_1*t + q_2*(1-t)$ trick.

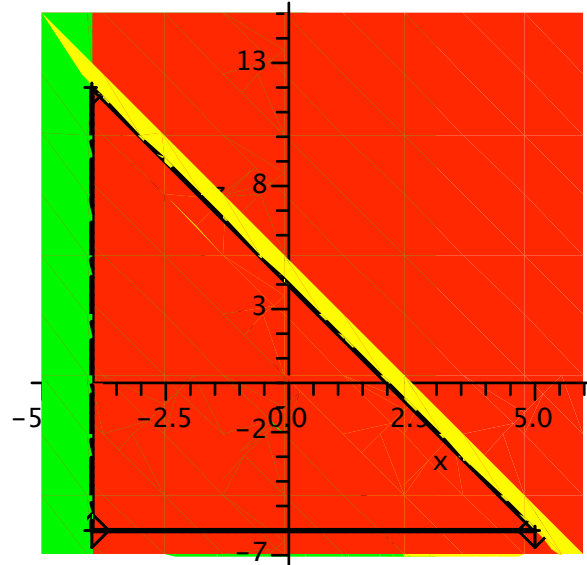
```
> s1 := plot3d([(y-4)*t+(-y/2+5)*(1-t), y, -6], t=0..1, y=0..6,  
transparency=.5, color=blue, style=patchnogrid):  
display([points, lines, s1], axes=normal);
```



For the other three planes we want to look at the frame down the y axis (orientation=[270,90]).

```
> eqn2 := y=0;
   eqn3 := y-x=4;
   eqn4 := 2*x+y+z=4;
   otherplanes := implicitplot3d([eqn2, eqn3, eqn4], x=-5..6, y=
-1..7, z=-7..15,
   color=[red, green, yellow], style=patchnogrid,
   transparency=.5, axes=normal);
   display([points, lines, otherplanes], axes=normal,
orientation=[270,90]);

      eqn2 := y=0
      eqn3 := y - x=4
      eqn4 := 2 x + y + z=4
```



The lines of the tetrahedron form 3 triangles (left, right, and outer) that all have z going from -6 to 12 .

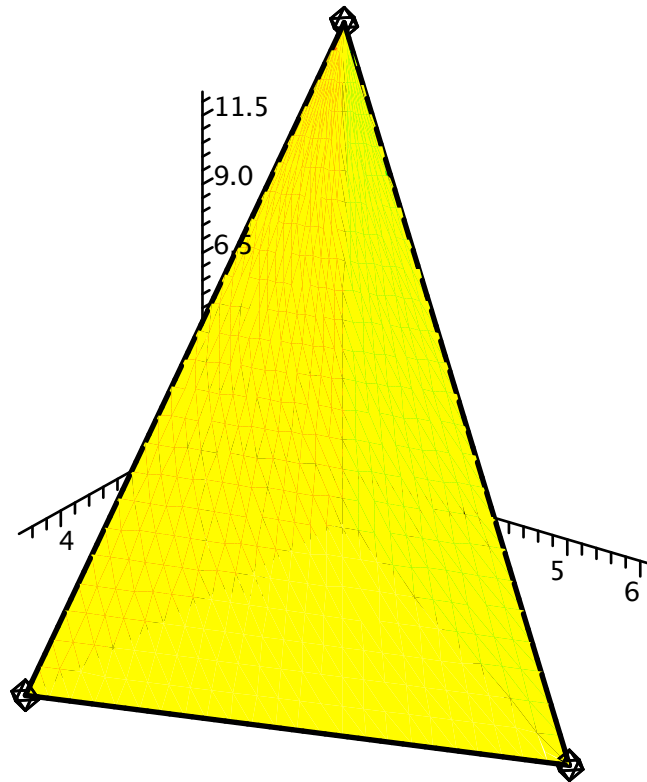
The outer triangle (red plane with $y=0$) has x going from $x=-4$ to $x=-z/2+2$.

The left triangle (green plane with $y=x+4$) has x going from $x=-4$ to $x=-3/z$.

The right triangle (yellow plane with $y=-2x-z+4$) has x going from $x=-z/3$ to $-z/2+2$.

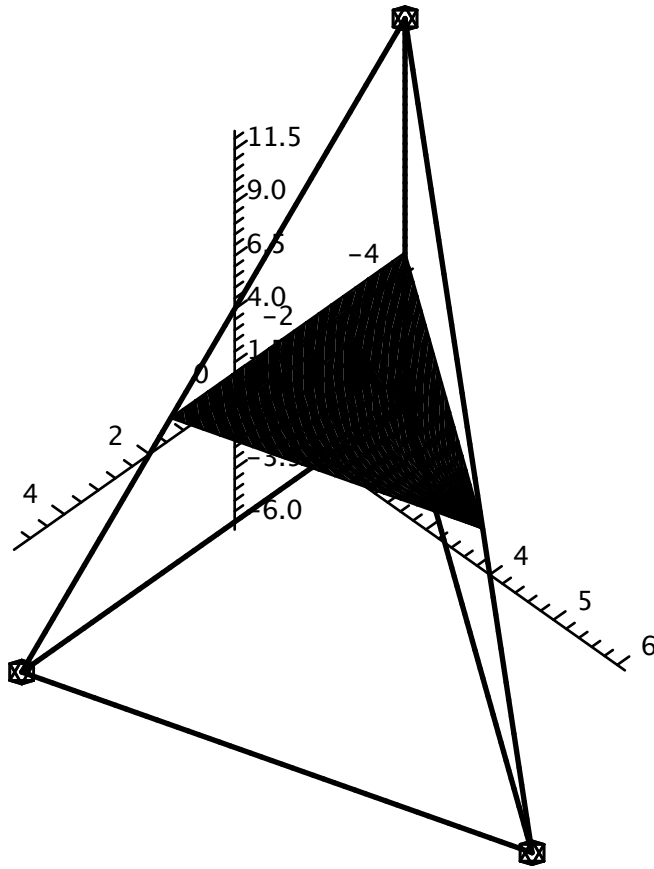
This lets us do our normal parameterization trick.

```
> s2 := plot3d([-4*t+(-z/2+2)*(1-t), 0, z], t=0..1, z=-6..12,
  transparency=.5, color=red, style=patchnogrid):
s3 := plot3d([-4*t+(-z/3)*(1-t), -4*t+(-z/3)*(1-t)+4, z],
  t=0..1, z=-6..12, transparency=.5, color=green, style=
  patchnogrid):
s4 := plot3d([-z/3*t+(-z/2+2)*(1-t), -2*(-z/3*t+(-z/2+2)*(1-
  t)) -z+4, z],
  t=0..1, z=-6..12, transparency=.5, color=yellow, style=
  patchnogrid):
display([points, lines, s2, s3, s4], axes=normal);
```



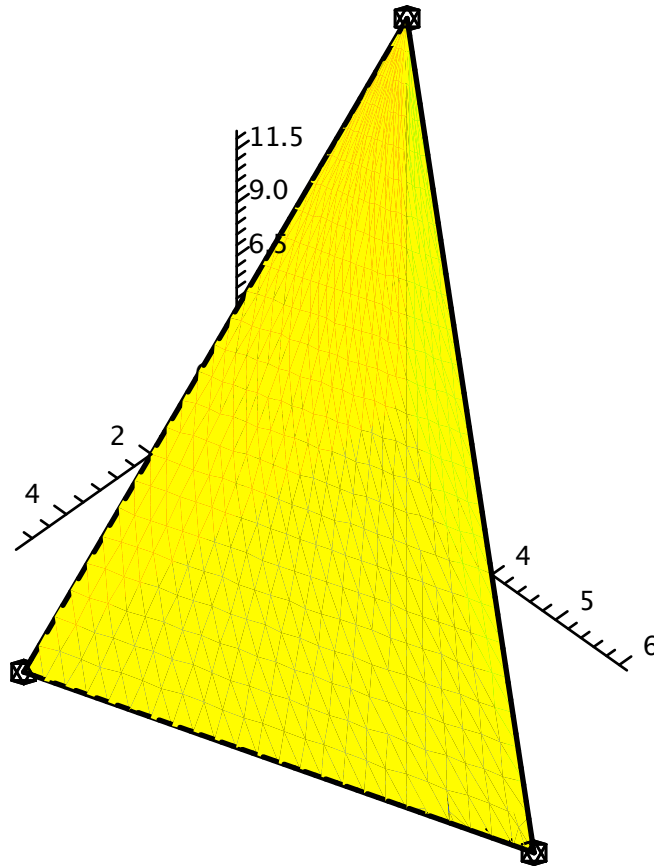
We can also look at a slice with z_0 set somewhere between -6 and 12. This is a mix of $(z_0+6)/18$ parts vertex = $(-4,0,12)$ and $(12-z_0)/18$ parts yellow triangle = $\langle (y-4)*t+(-y/2+5)*(1-t), y, -6 \rangle$.

```
> z0:=1;
s5 := plot3d(<-4,0,12>*(z0+6)/18+<(y-4)*t+(-y/2+5)*(1-t),y,
-6>*(12-z0)/18,
t=0..1,y=0..6, transparency=0,color=black,grid=[20,20]
):
display([points, lines, s5],axes=normal);
z0:=1
```



With a little work we can make this an animation of the slices.

```
> z0 := 'z0':
  slice := z0 ->
    plot3d(<-4,0,12>*(z0+6)/18+<(y-4)*t+(-y/2+5)*(1-t),y,-6>*
      (12-z0)/18,
      t=0..1,y=0..6, transparency=0,color=black,grid=[20,20]
      ,axes=normal):
> display([seq(display([points, lines, slice(ht), s1, s2, s3,
  s4]),ht=-6..12)],
  insequence=true,axes=normal);
```



To view the animation, select the picture, reduce the frame rate to 2 per second, set the loop to repeat, and start the animation.

It may be clearer to look at the animation without the sides.

```
> display([seq(display([points, lines, slice(ht)]),ht=-6..12)],  
insequence=true,axes=normal);
```

