Homework 1

By Russell Blyth

```
> restart: with(LinearAlgebra): with(plots): with(plottools):with
  (geom3d):
Warning, the name changecoords has been redefined
Warning, the assigned name arrow now has a global binding
Warning, these names have been rebound: dodecahedron, hexahedron,
homothety, icosahedron, line, octahedron, parallelepiped, point,
reflect, rotate, sphere, stellate, tetrahedron, transform, translate
Warning, the assigned name polar now has a global binding
```

The line of intersecton of two planes

Define three planes to work with.

```
> eq10 := 2*x + 3*y + z = 2;
  eq11 := 2*x + 4*y + 7*z =6;
  eq12 := 3*x + 10*y + 5*z = 7;
  solve({eq10,eq11,eq12});
```

$$eq10 := 2\,x + 3\,y + z = 2$$

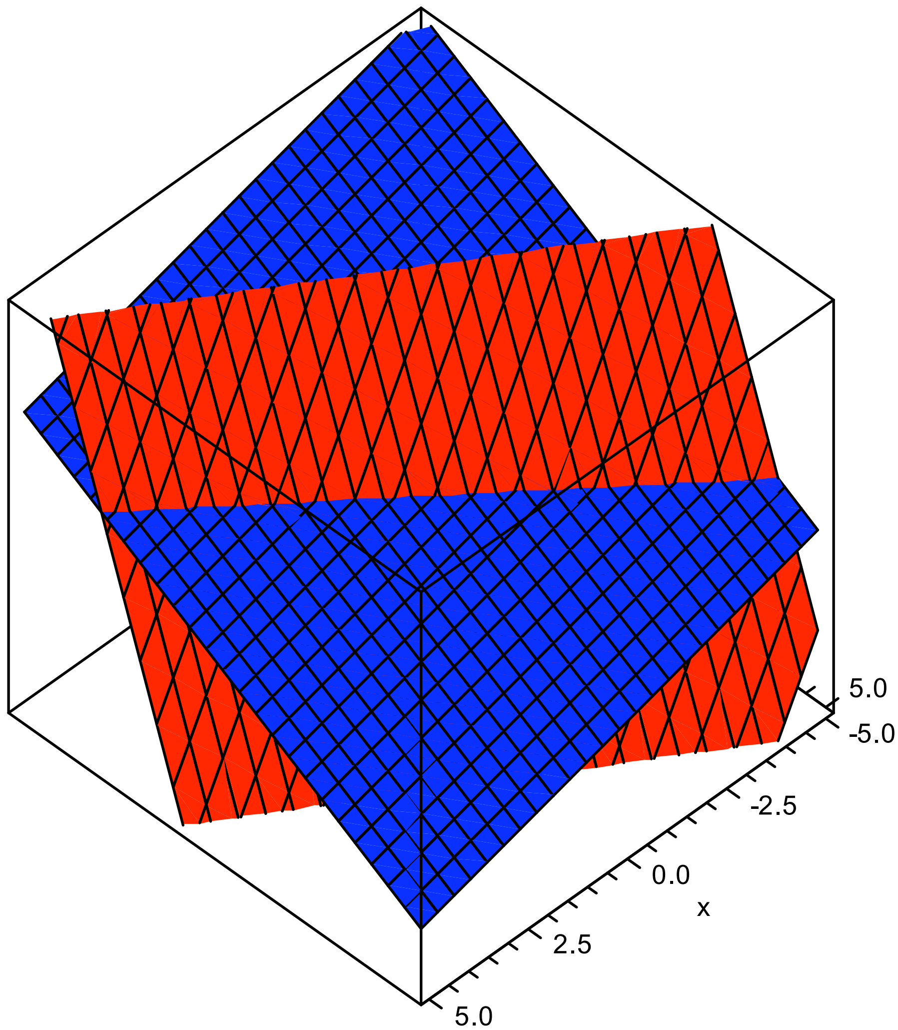
$$eq11 := 2\,x + 4\,y + 7\,z = 6$$

$$eq12 := 3\,x + 10\,y + 5\,z = 7$$

$$\left\{x = \frac{11}{59}, y = \frac{20}{59}, z = \frac{36}{59}\right\} \quad (1.1)$$

Graph two planes.

```
> display({
  plot3d(solve(eq10,z),x=-5..5, y=-5..5, color=red),
  plot3d(solve(eq11,z),x=-5..5, y=-5..5, color=blue)},
  view=-5..5, axes=boxed);
```



Now we want to display the line of intersection in a heavy line. First find a solution for the intersection. Use commands from the geometry package.

```
> plane(pl1,eq10,[x,y,z]):
  plane(pl2,eq11,[x,y,z]):
  intersection(ln1,pl1,pl2);
```

$$ln1 \tag{1.2}$$

Extract the equation of the line.

```
> LineEq := Equation(ln1,t);
```

$$LineEq := [-5 + 17\,t, 4 - 12\,t, 2\,t] \tag{1.3}$$

The tricky part is determining values for the parameter t that will plot the part of the line that corresponds exactly to the parts of the planes that are displayed. The following code does that.

```
> xt1:=solve(LineEq[1]=-5,t);
  xt2:=solve(LineEq[1]=5,t);
  yt1:=solve(LineEq[2]=-5,t);
  yt2:=solve(LineEq[2]=5,t);
```

$$xt1 := 0$$

$$xt2 := \frac{10}{17}$$

$$yt1 := \frac{3}{4}$$

$$yt2 := \frac{-1}{12} \quad (1.4)$$

```
> xtm := min(xt1,xt2);
  xtM := max(xt1,xt2);
  ytm := min(yt1,yt2);
  ytM := max(yt1,yt2);
  tm  := max(xtm,ytm);
  tM  := min(xtM,ytM);
```

$$xtm := 0$$

$$xtM := \frac{10}{17}$$

$$ytm := \frac{-1}{12}$$

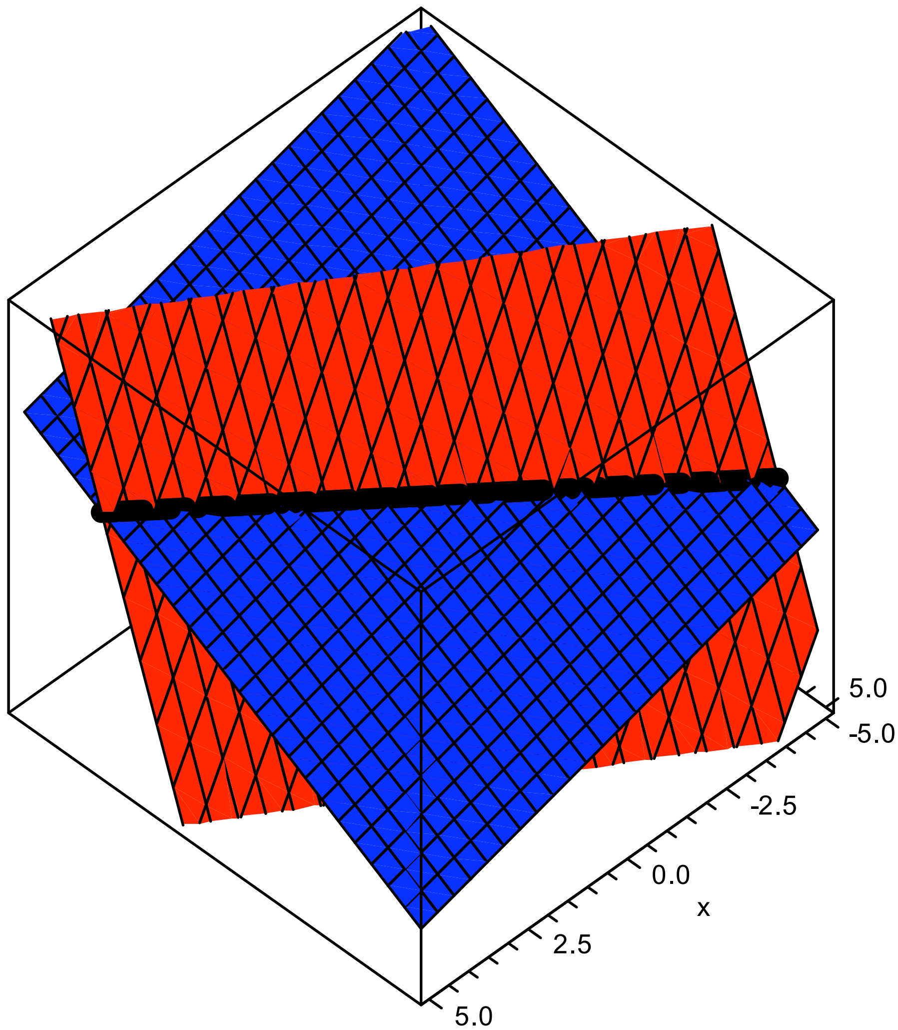
$$ytM := \frac{3}{4}$$

$$tm := 0$$

$$tM := \frac{10}{17} \quad (1.5)$$

Now graph the planes and the line using t=tm..tM

```
> display({
  plot3d(solve(eq10,z),x=-5..5, y=-5..5, color=red),
  plot3d(solve(eq11,z),x=-5..5, y=-5..5, color=blue),
  spacecurve(Equation(ln1,t),t=tm..tM, color=black,thickness=8)},
  view=-5..5, axes=boxed);
```



Pack this into a procedure and use it for the other two possible intersections of the three planes. The parameters for the procedure IntLine are the equations of the lines, along with the intended intervals for plotting, x=xm..xM, y=ym,yM and the plot color for the line of intersection.

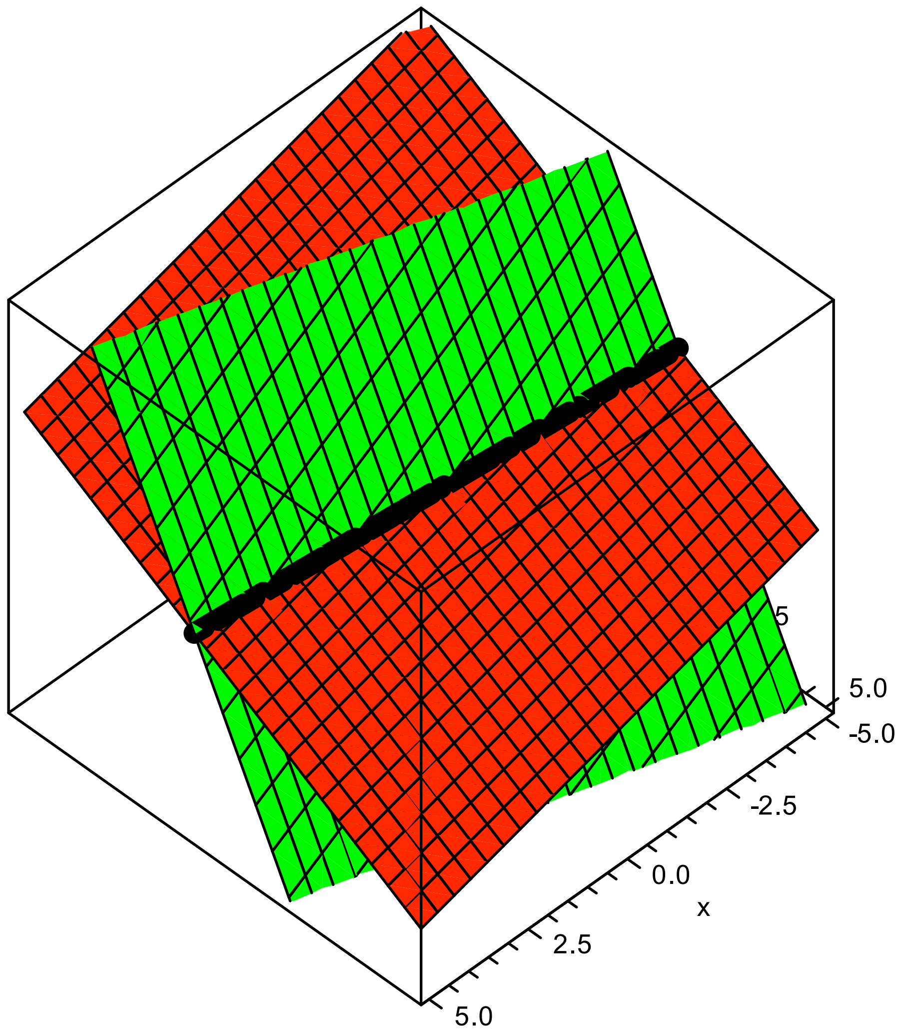
```
> IntLine := proc(eq1, eq2,xm,xM,ym,yM,pc)
    local LineEq, xt1,xt2,yt1,yt2, xtm,xtM,ytm,ytM,tm,tM:
    plane(pl1,eq1,[x,y,z]):
    plane(pl2,eq2,[x,y,z]):
    intersection(ln1,pl1,pl2):
    LineEq := Equation(ln1,t):
    xt1:=solve(LineEq[1]=xm,t):
    xt2:=solve(LineEq[1]=xM,t):
    yt1:=solve(LineEq[2]=ym,t):
    yt2:=solve(LineEq[2]=yM,t):
    xtm := min(xt1,xt2):
    xtM := max(xt1,xt2):
    ytm := min(yt1,yt2):
    ytM := max(yt1,yt2):
    tm := max(xtm,ytm):
    tM := min(xtM,ytM):
    spacecurve(Equation(ln1,t),t=tm..tM, color=pc, thickness=8):
  end proc;
```

IntLine := proc (eq1, eq2, xm, xM, ym, yM, pc) (1.6)

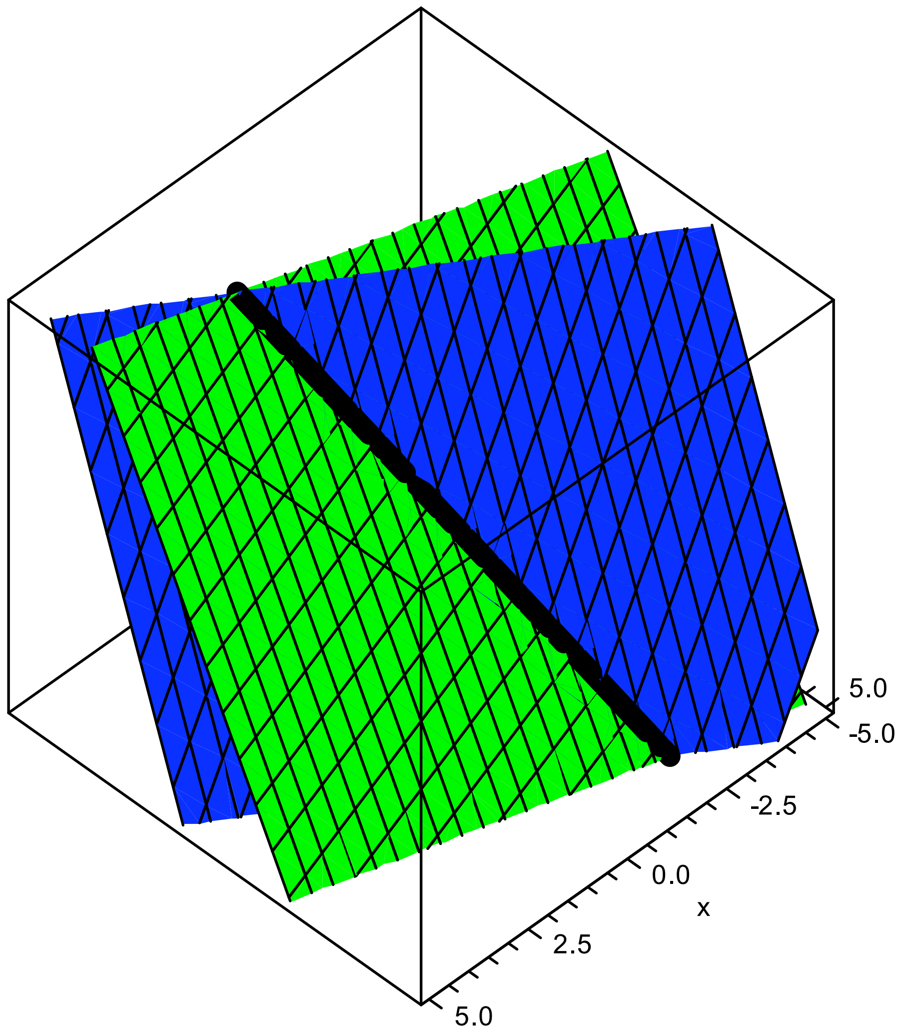
```
local LineEq, xt1, xt2, yt1, yt2, xtm, xtM, ytm, ytM, tm, tM;
(geom3d:-plane)(pl1, eq1, [x, y, z]);
(geom3d:-plane)(pl2, eq2, [x, y, z]);
(geom3d:-intersection)(ln1, pl1, pl2);
LineEq := (geom3d:-Equation)(ln1, t);
xt1 := solve(LineEq[1] = xm, t);
xt2 := solve(LineEq[1] = xM, t);
yt1 := solve(LineEq[2] = ym, t);
yt2 := solve(LineEq[2] = yM, t);
xtm := min(xt1, xt2);
xtM := max(xt1, xt2);
ytm := min(yt1, yt2);
ytM := max(yt1, yt2);
tm := max(xtm, ytm);
tM := min(xtM, ytM);
spacecurve((geom3d:-Equation)(ln1, t), t = tm..tM, color = pc, thickness = 8)
end proc
```

We can now plot a pair of planes and the line of intersection.

```
> display({
plot3d(solve(eq11,z),x=-5..5, y=-5..5, color=red),
plot3d(solve(eq12,z),x=-5..5, y=-5..5, color=green),
IntLine(eq11,eq12,-5,5,-5,5,black)},
view=-5..5, axes=boxed);
```

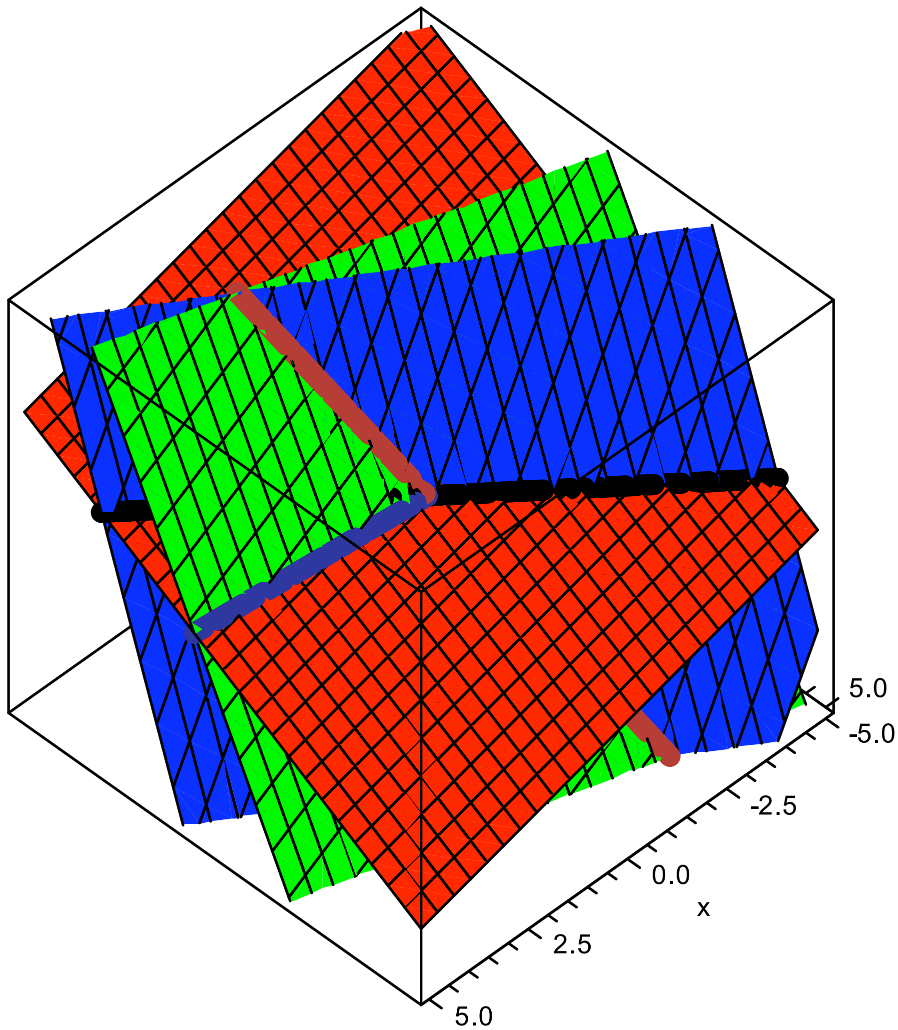



```
> display({
  plot3d(solve(eq10,z),x=-5..5, y=-5..5, color=blue),
  plot3d(solve(eq12,z),x=-5..5, y=-5..5, color=green),
  IntLine(eq10,eq12,-5,5,-5,5,black)},
  view=-5..5, axes=boxed);
```



and one final test:

```
> display({
  plot3d(solve(eq10,z),x=-5..5, y=-5..5, color=blue),
  plot3d(solve(eq11,z),x=-5..5, y=-5..5, color=red),
  plot3d(solve(eq12,z),x=-5..5, y=-5..5, color=green),
  IntLine(eq10,eq11,-5,5,-5,5,black),
  IntLine(eq10,eq12,-5,5,-5,5,brown),
  IntLine(eq11,eq12,-5,5,-5,5,navy)},
  view=-5..5, axes=boxed);
```


5.0
-5.0
-2.5
0.0
x
2.5
5.0

>