Three-Dimensional Viewing and Clipping

1----------

In[22]:=
```
unitCube = {{0,0,0,1},{1,0,0,1},{1,1,0,1},{0,1,0,1},{0,1,1,1},{0,0,1,1},{1,0,1,1},{1,1
```

In[30]:=
```
Transpose[unitCube] // MatrixForm
```

Out[30]//MatrixForm=

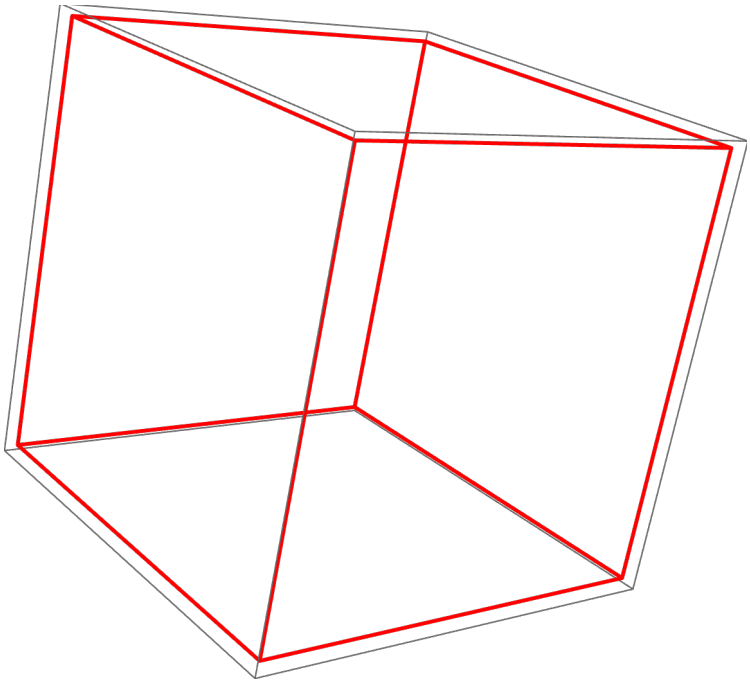
$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

In[109]:=
```
edges = {
    {1, 2}, {2, 3}, {3, 4}, {4, 1},
    {5, 6}, {6, 7}, {7, 8}, {8, 5},
    {1, 6}, {2, 7}, {3, 8}, {4, 5}
};

lines = Line /@ (unitCube[[#]][[All, ;; 3]] & /@ edges);

pPlot = Graphics3D[{Red, Thick, lines},
  AxesOrigin → {0, 0, 0},
  AxesLabel → {"X", "Y", "Z"}]
```

Out[111]=



In[26]:=
```
perK = {{d,0,0,0}, {0,d,0,0},{0,0,0,0},{0,0,1,d}};
```

In[27]:=
```
perK // MatrixForm
```

Out[27]//MatrixForm=

$$\begin{pmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & d \end{pmatrix}$$

In[79]:=
```
pD1 = perK.Transpose[unitCube] /. d→1;
```

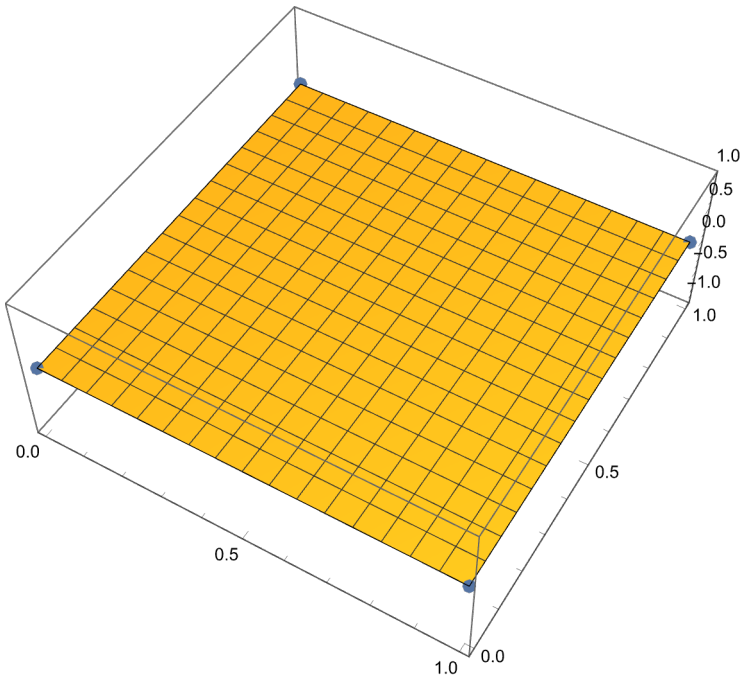
In[80]:=
```
pD1 // MatrixForm
```

Out[80]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \end{pmatrix}$$

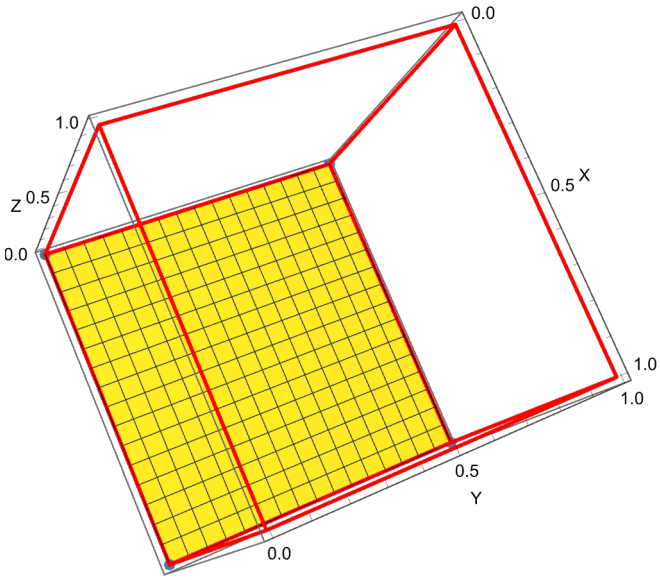
In[95]:=
```
pD1Plot = Show[ListPlot3D[{Transpose[pD1][[All,;;3]]}], ListPointPlot3D[Transpose[pD1][[A
```

Out[95]=



In[113]:=
```
Show[pD1Plot,pPlot,  PlotRange→All, AxesLabel→{"X", "Y", "Z"}, BoxRatios → {1, 1, 1}]
```

Out[113]=

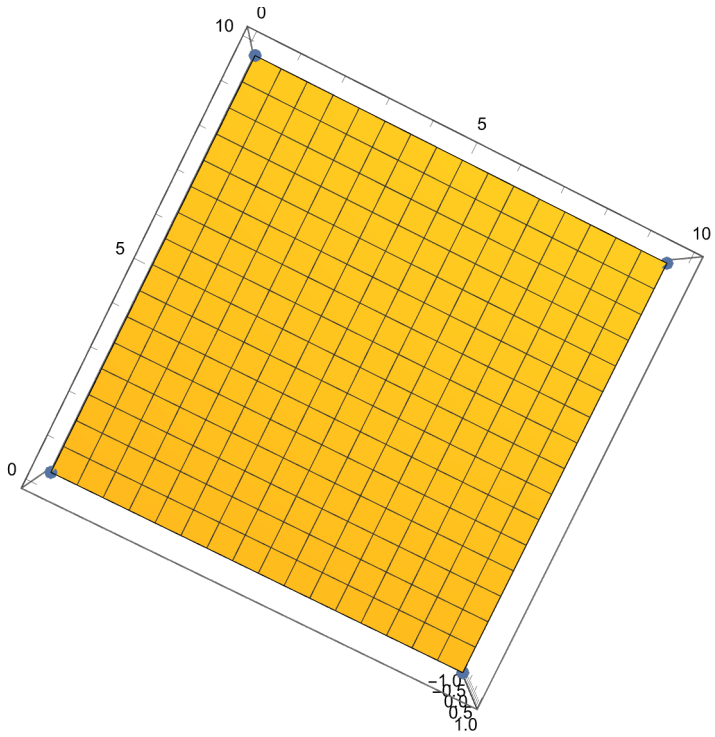


In[114]:=
```
pD10 = perK.Transpose[unitCube] /. d→10;
```

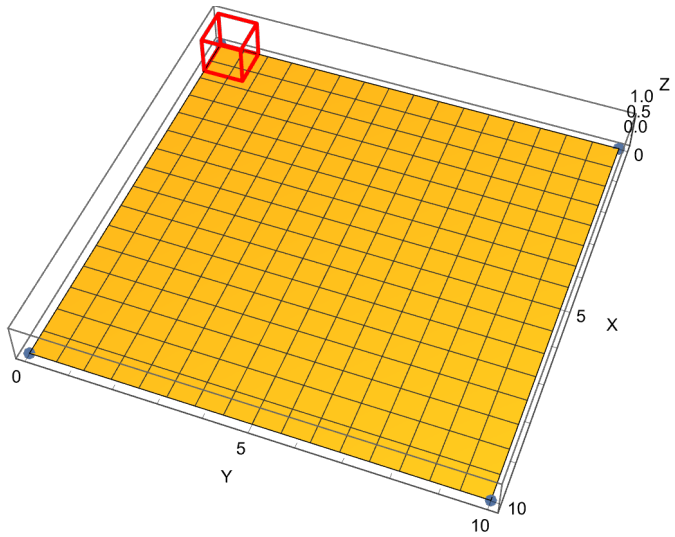
In[115]:=
```
pD10 // MatrixForm
```
Out[115]//MatrixForm=

$$\begin{pmatrix} 0 & 10 & 10 & 0 & 0 & 0 & 10 & 10 \\ 0 & 0 & 10 & 10 & 10 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 10 & 10 & 10 & 10 & 11 & 11 & 11 & 11 \end{pmatrix}$$

In[117]:=
```
pD10Plot = Show[ListPlot3D[{Transpose[pD10][[All,;;3]]}], ListPointPlot3D[Transpose[pD10
```
Out[117]=



In[122]:=
```
Show[pD10Plot, pPlot,  PlotRange→All, AxesLabel→{"X", "Y", "Z"}, BoxRatios → {1, 1, 0.
```
Out[122]=



2----------

In[128]:=
```
pPrime[p_,d_] := Return[{(d*p[[1]])/(p[[3]]+d), (d*p[[2]])/(p[[3]]+d), 0}]
```

In[129]:=
```
p0 = {1,1,-4};
p1 = {2,4,4};
```

In[132]:=
```
COP = {0,0,-1};
```

Answer 2.a

In[134]:=
```
p0Prime = pPrime[p0,-COP[[3]]]
p1Prime = pPrime[p1,-COP[[3]]]
```

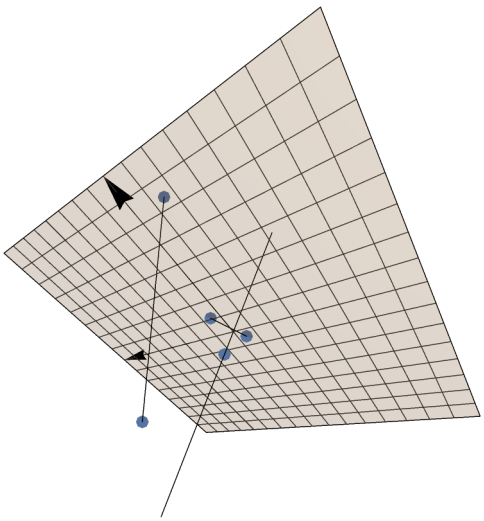
Out[134]=
$$\left\{-\frac{1}{3}, -\frac{1}{3}, 0\right\}$$

Out[135]=
$$\left\{\frac{2}{5}, \frac{4}{5}, 0\right\}$$

In[171]:=

```
perView = Show[
    Graphics3D[
        {Arrow[{{0, 0, 0}, {6, 0, 0}}],
        Arrow[{{0, 0, 0}, {0, 6, 0}}],
        Line[{{0, 0, 6}, {0, 0, -6}}]},
        AxesOrigin → {0, 0, 0},
        Boxed→False
    ],
    ListPointPlot3D[{p0, p1, COP}],
    Graphics3D[Line[{p0, p1}]],
    ListPointPlot3D[{p0Prime, p1Prime}],
    Graphics3D[Line[{p0Prime, p1Prime}]],
    Plot3D[0, {x, -6, 6}, {y, -6, 6}, PlotStyle → Opacity[0.2]],
    PlotRange→All
]
```

Out[171]=



In[189]:=

```
x = p0⟦1⟧ + T(p1⟦1⟧-p0⟦1⟧);
y = p0⟦2⟧ + T(p1⟦2⟧-p0⟦2⟧);
z = p0⟦3⟧ + T(p1⟦3⟧-p0⟦3⟧);
```

In[192]:=

```
(*find t by given z = 0*)
t = -p0⟦3⟧/(p1⟦3⟧-p0⟦3⟧)
```

Out[192]=

$$\frac{1}{2}$$

Answer 2.b

In[188]:=

```
interceptZ0 = {x,y,z} /. T → t
```

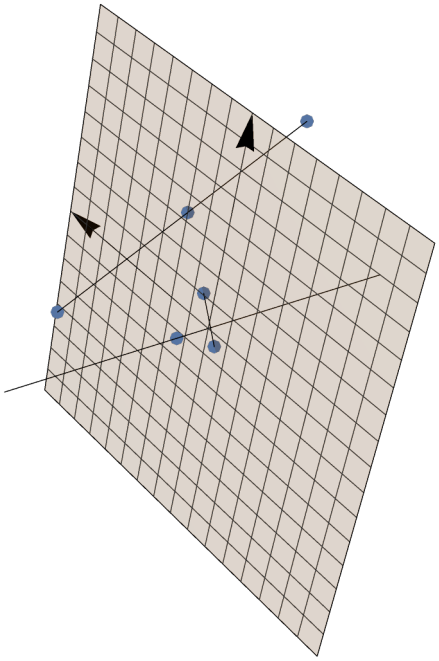
Out[188]=

$$\left\{\frac{3}{2}, \frac{5}{2}, 0\right\}$$

In[193]:=

```
Show[
    perView,
    ListPointPlot3D[{interceptZ0}]
]
```

Out[193]=



In[196]:=

```
p2 = {3,1,6};
p2Prime = pPrime[p2,-COP[[3]]]
```

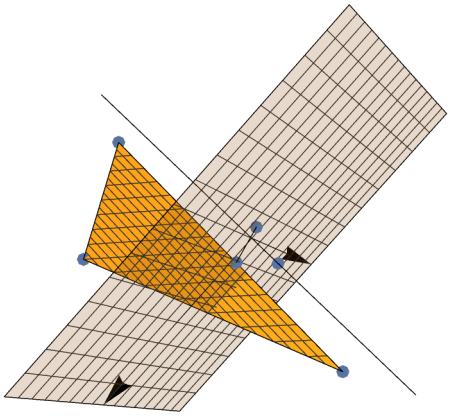
Out[197]=

$$\left\{\frac{3}{7}, \frac{1}{7}, 0\right\}$$

In[199]:=

```
Show[
    perView,
    ListPointPlot3D[{p2}],
    ListPlot3D[{p0,p1,p2}]
]
```

Out[199]=



In[200]:=

```
FindInterceptZ0[p0_,p1_]:= Module[{t},
    t = -p0⟦3⟧/(p1⟦3⟧-p0⟦3⟧);
    Return[{p0⟦1⟧ + t(p1⟦1⟧-p0⟦1⟧), p0⟦2⟧ + t(p1⟦2⟧-p0⟦2⟧), 0}]
]
```

Answer 2.c

In[211]:=

```
PolygonClipping[p_] := Module[{i, k, iP, iK, newPoints = {}, intercept},
  For[i = 1, i ≤ Length[p], i++,
    k = Mod[i, Length[p]] + 1;
    iP = p[[i]];
    iK = p[[k]];

    If[iP[[3]] * iK[[3]] < 0,
      intercept = FindInterceptZ0[iP, iK];
      AppendTo[newPoints, intercept];
    ];
    If[iP[[3]] ≥ 0, AppendTo[newPoints, iP]];
  ];
  Return[newPoints];
];
clippedPoints = N[PolygonClipping[{p0,p1,p2}]]
```

Out[212]=

```
{{1.5, 2.5, 0.}, {2., 4., 4.}, {1.8, 1., 0.}, {3., 1., 6.}}
```

In[210]:=

```
Show[
    perView,
    ListPointPlot3D[{p0, p1, p2}, PlotStyle → Red],
    Graphics3D[Line[{p0, p1, p2, p0}]],
    ListPlot3D[clippedPoints]
]
```

Out[210]=