

# 3D to 2D

vertex
+ x - coordinate : Float
+ y - coordinate : Float
+ z - coordinate : Float
+ Transformation [ int int int ] : vertex

Vertex - A

Projection on xy-plane  
multiply with [ 1 1 0 ]

Vertices of  
Front View

Repeat this for each vertex

Projection on yz-plane  
multiply with [ 0 1 1 ]

Projection on xz-plane  
multiply with [ 1 0 1 ]

Vertices of  
Side View

Vertices of  
Top View

Edge
+ vertex1 : Vertex
+ vertex2 : Vertex
+ transformation [ int int int ] () : Edge

Edge 1

Projection on xy-plane ->  
multiply given two  
vertices with [ 1 1 0 ]

Edge on  
Front View

Repeat this for each edge

Projection on yz-plane  
-> multiply given two  
vertices with [ 0 1 1 ]

Projection on xz-plane -  
> multiply given two  
vertices with [ 1 0 1 ]

Edge on  
Side View

Edge on  
Top View

Finally we have obtained all the vertices and edges of all three views

# 2D to 3D

vertex xy
+ x-coordinate : Float
+ y-coordinate : Float
+ Remark : this is a vertex in xy-plane

vertex yz
+ y-coordinate : Float
+ z-coordinate : Float
+ Remark : this is a vertex in yz-plane

vertex xz
+ x-coordinate : Float
+ z-coordinate : Float
+ Remark : this is a vertex in xz-plane

Vertes xy - A(x1, y1)

check for all vertices in yz-plane  
having y1 as y-coordinate

y1  
z1

y1  
z2

y1  
zn

Check whether (x1,z1) is in  
the xz plane

Yes

(x1,y1,z1) is a  
Vertex of a 3D  
object

No

(x1,y1,z1) is  
not a Vertex of  
a 3D object

Check for all the Vertices-xy  
(x2,y2), (x3,y3)----- (xn,yn)

Check for all the Points (x1,z2),  
(x1,z3)----- (x1,zn)

now we got all the 3D vertices

Edge
+ vertex1 : Vertex
+ vertex2 : Vertex
+

Edge between  
vertex1(x1, y1,z1)  
vertex2(x2,y2,z2)

check whether(x1,y1) and  
(x2,y2) have an edge  
between them in xy plane

yes

check whether(y1,z1) and  
(y2,z2) have an edge  
between them in yz plane

check whether(x1,z1) and  
(x2,z2) have an edge  
between them in xz plane

yes

then there is  
edge between  
(x1,y1,z1) and  
(x2,y2,z2)

similarly check for all the vertices pairs