

UNIT CUBE OPENGL

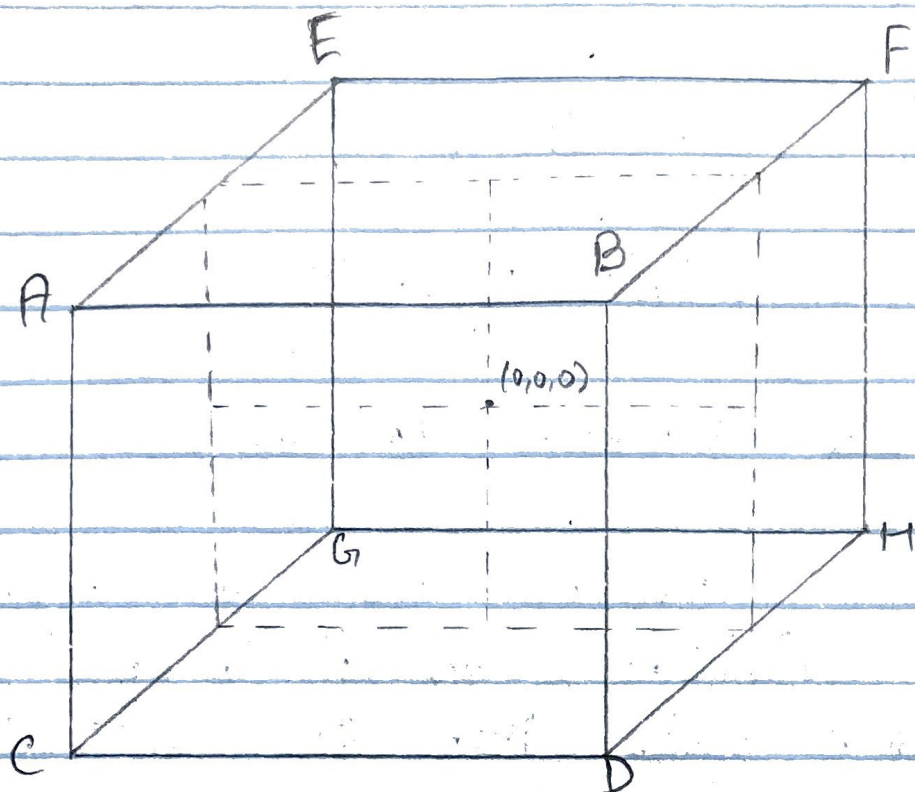


Fig: Unit cube

A unit cube has:

8 distinct positions

A $(-.5, .5, .5)$

B $(.5, .5, .5)$

C $(-.5, -.5, .5)$

D $(.5, -.5, .5)$

E $(-.5, .5, -.5)$

F $(.5, .5, -.5)$

G $(-.5, -.5, -.5)$

H $(.5, -.5, -.5)$

6 distinct faces / normals

ACDB $(0, 0, 1)$

EFHG $(0, 0, -1)$

BDHF $(1, 0, 0)$

GHDG $(0, -1, 0)$

EABF $(0, 1, 0)$

AEHC $(-1, 0, 0)$

24 position attributes in OpenGL

24 normal attributes in OpenGL

36 index elements in OpenGL

Here we are specifying our cube with basic triangle primitive, "GL_TRIANGLES" in OpenGL draw calls. There are 12 total triangles to be drawn. With consideration to default counter clockwise front facing winding order of OpenGL, the above cube could be expressed in following order:

(CCW) Counter clockwise faces: (visible)

Quads

AEDB

BDHF

EABF

Triangles

CBA and CDB

DFB and DHF

AFE and ABF

(CW) Clockwise faces: (culled)

Quads

AEGC

EFHC

GNDC

Triangles

CAE and CEG

GEF and GHF

GNH and GND

Vertex Attributes are then derived from the CW and CCW quad faces:

Attribute Index	Position Attrib	Normal Attrib
0	A	$(0, 0, 1)$
1	C	$(0, 0, 1)$
2	D	$(0, 0, 1)$
3	B	$(0, 0, 1)$
4	B	$(1, 0, 0)$
5	D	$(1, 0, 0)$
6	H	$(1, 0, 0)$
7	F	$(1, 0, 0)$
8	E	$(0, 1, 0)$
9	A	$(0, 1, 0)$
10	B	$(0, 1, 0)$
11	F	$(0, 1, 0)$
12	A	$(-1, 0, 0)$
13	E	$(-1, 0, 0)$
14	G	$(-1, 0, 0)$
15	C	$(-1, 0, 0)$
16	E	$(0, 0, -1)$
17	F	$(0, 0, -1)$
18	H	$(0, 0, -1)$
19	G	$(0, 0, -1)$
20	G	$(0, -1, 0)$
21	H	$(0, -1, 0)$
22	D	$(0, -1, 0)$
23	C	$(0, -1, 0)$

Index elements are derived from CW and CCW triangle faces and attribute indices from the table above

Index Elements	Triangle	Winding Order
1, 3, 0	CBA	CCW
1, 2, 3	CDB	CCW
5, 7, 4	DEB	CCW
5, 6, 7	DHF	CCW
9, 11, 8	AFF	CCW
9, 10, 11	ABF	CCW
15, 12, 13	CAE	CW
15, 13, 14	CEG	CW
19, 16, 17	GFE	CW
19, 17, 18	GFI	CW
23, 20, 21	CGH	CW
23, 21, 22	CHD	CW

In total 36 index Elements