

```
import pygame
from pygame.locals import *
```

```
from OpenGL.GL import *
from OpenGL.GLU import *
```

```
vertices = (
    # ( x,  y,  z)
    ( 1, -1, -1), # A
    ( 1,  1, -1), # B
    (-1,  1, -1), # C
    (-1, -1, -1), # D
    ( 1, -1,  1), # E
    ( 1,  1,  1), # F
    (-1, -1,  1), # G
    (-1,  1,  1)  # H
)
```

```
edges = (
    (0, 1),
    (0, 3),
    (0, 4),
    (2, 1),
    (2, 3),
    (2, 7),
    (6, 3),
    (6, 4),
    (6, 7),
    (5, 1),
    (5, 4),
    (5, 7),
    # (7, 1),
    # (5, 2),
    # (1, 4),
    # (5, 0),
    # (7, 4),
    # (6, 5),
    # (6, 2),
    # (7, 3),
    # (2, 0),
    # (1, 3),
    # (6, 0),
    # (3, 4),
    # (7, 0),
    # (5, 3),
    # (2, 4),
```

```

    # (1, 6)
)

surfaces = (
    (0, 1, 2, 3),
    (3, 2, 7, 6),
    (6, 7, 5, 4),
    (4, 5, 1, 0),
    (1, 5, 7, 2),
    (4, 0, 3, 6)
)

color = (
    (1, 0, 0),
    (0, 1, 0),
    (0, 0, 0),
    (1, 1, 1),
    (0, 1, 1),
    (0, 1, 1),
    (1, 0, 0),
    (0, 1, 0),
    (0, 0, 1),
    (0, 1, 0),
    (0, 0, 1),
    (0, 1, 0),
)

def Cube():
    glBegin(GL_QUADS)
    for surface in surfaces:
        x = 0
        glColor3fv((1, 0, 0))
        for vertex in surface:
            x += 1
            glColor3fv(color[x])
            glVertex3fv(vertices[vertex])
    glEnd()

    glBegin(GL_LINES)
    glColor3fv((0, 0.9, 0))
    for edge in edges:
        for vertex in edge:
            glVertex3fv(vertices[vertex])
    glEnd()

def main():

```

```
pygame.init()
screen = pygame.display.set_mode((800, 600), DOUBLEBUF | OPENGGL)
```

```
gluPerspective(45, (800 / 600), 0.1, 50)
glTranslatef(0, 0, -5)
glRotatef(0, 0, 0, 0)
```

```
while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()

    glRotatef(1, 1, 1, 1)
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
    Cube()
    pygame.display.flip()
    pygame.time.wait(10)
```

```
main()
```