

LAPORAN AKHIR SEMESTER
MATA KULIAH GRAFIKA KOMPUTER
“MENG GAMBAR LIMAS 3D”



Oleh:

MEGA INTAN PRATIWI

(20051397007)

Link Github :

<https://github.com/megaintan007/Grafika-Komputer.git>

KELAS 2020 A
D4 MANAJEMEN INFORMATIKA
PROGRAM VOKASI
UNIVERSITAS NEGERI SURABAYA

2022

- PENJELASAN SINGKAT BANGUN RUANG LIMAS SEGI EMPAT

- Install puOpenGL dan pygame sebelum membuat bangun ruang 3D, kemudian import dan tulis semua modul tersebut ke dalam aplikasi visual studio code agar objek animasi 3D dapat muncul.

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

- **Vertices (Titik-titik)** : sebuah fungsi untuk menentukan/menggambar titik koordinat pada saat membuat bangun ruang. Vertices ini mempunyai 3 sumbu yaitu sumbu X, Y, Z. Dalam limas segi empat mempunyai 5 titik, maka perlu didefinisikan menjadi 5 vertex seperti dibawah ini :

```
#mendefinisikan sudut-sudutnya
vertices = (
#Vertices Rotasi Tidur
(1, 0, -1), #Titik 0
(1, 0, 1), #Titik 1
(-1, 0, 1), #Titik 2
(-1, 0, -1), #Titik 3
(0, 3, 0), #Titik 4
)
```

- **Edge** : fungsi untuk menentukan “tepi” dari bangun ruang limas segi empat. Masing-masing tepi berisi 2 angka. Angka-angka tersebut mengacu pada vertex yang ada diatas, kemudian “tepi” tersebut akan ditarik di antara kedua vertex.

```
#mendefinisikan garis
edges = (
    (0, 1),
    (0, 3),
    (0, 4),
    (2, 1),
    (2, 3),
    (2, 4),
    (1, 4),
    (3, 4),
)
```

- Selanjutnya kita beri warna pada limas segi empat agar tampilannya lebih menarik dan terlihat seperti objek 3D. Untuk mendefinisikan/memberi warna pada bangun ruang limas segi empat kita perlu menggunakan “**glcolors**”

```
#mendefinisikan warna
colors = (
    (1, 1, 1),
    (1, 0, 0),
    (1, 1, 0),
    (0, 0, 1),
    (1, 0, 1),
    (0, 1, 1),
    (1, 1, 1),
    (1, 0, 0),
    (1, 1, 0),
    (0, 0, 1),
    (1, 0, 1),
    (0, 1, 1),
)
```

- Kemudian kita mendefinisikan **surface (sisi)** limas segi empat. Penjelasan sisi limas segi empat dapat kita lihat dalam source code berikut ini :

```
#mendefinisikan sisi
surfaces = (
    #Diamond
    (0, 1, 4),
    (0, 3, 4),
    (2, 1, 4),
    (2, 3, 4),
)
```

- Setelah semuanya sudah di definisikan, selanjutnya kita mulai untuk menggambar limas segi empat nya dengan cara sebagai berikut :

```
#menggambar Limas
def Limastumpuk():
    glBegin(GL_LINES)
    for edge in edges:
        for vertex in edge:
            glVertex3fv(vertices[vertex])
    glEnd()

    glBegin(GL_TRIANGLES)
    for surface in surfaces:
        x = 0
        for vertex in surface:
```

```

        x += 1
        glColor3fv(colors[x])
        glVertex3fv(vertices[vertex])
    glEnd()

```

- Untuk menampilkan objek 3D limas segi empat kita perlu mendefinisikannya sebagai berikut :

```

def main():
    #inisialisasi pygame
    pygame.init()
    #resolusi display layar
    display = (800,600)
    #mode layar double buffering
    pygame.display.set_mode(display, DOUBLEBUF|OPENGL)

```

- Kita juga bisa mengatur nama judul pada window, yaitu dengan cara :

```

#mengatur judul pada windows
pygame.display.set_caption("Menggambar Limas - Mega")

```

- Untuk memindahkan posisi objek pada sumbu X, Y, Z kita perlu mendefinisikannya sebagai berikut :

```

#memindahkan objek sesuai dengan matrix translate
glTranslatef(0.0,0.0, -7)

```

- Untuk mengatur agar program tidak berjalan secara terus-menerus pada saat kita ingin menutup output, kita perlu mendefinisikannya sebagai berikut :

```

#infinite looping
while True:
    #apabila ditekan tombol x, maka program akan berhenti.
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            quit()

```

- Selanjutnya kita mengatur perputaran posisi limas. Parameter yang digunakan adalah sumbu X, Y, Z. Apabila parameter 0 bernilai positif, maka objek akan berputar berlawanan arah jarum jam. Sedangkan jika parameter 0 bernilai negatif, maka objek akan berputar searah jarum jam

```

#arah gerak perputaran limas
glRotatef(1, 1, 1, 0)

```

- glClear berfungsi untuk menghapus semua display/kanvas. Setelah kanvas bersih kita bisa memanggil fungsi Limas()

```
#menghapus semua kanvas/display  
glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)  
Limastumpuk()  
pygame.display.flip()
```

- SOURCE CODE

Link Github : <https://github.com/megaintan007/Grafika-Komputer.git>

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

#mendefinisikan sudut-sudutnya
vertices = (
#Vertices Rotasi Tidur
(1, 0, -1), #Titik 0
(1, 0, 1),  #Titik 1
(-1, 0, 1), #Titik 2
(-1, 0, -1), #Titik 3
(0, 3, 0), #Titik 4
)

#mendefinisikan garis
edges = (
    (0, 1),
    (0, 3),
    (0, 4),
    (2, 1),
    (2, 3),
    (2, 4),
    (1, 4),
    (3, 4),
)

#mendefinisikan warna
colors = (
    (1, 1, 1),
    (1, 0, 0),
    (1, 1, 0),
    (0, 0, 1),
    (1, 0, 1),
    (0, 1, 1),
    (1, 1, 1),
    (1, 0, 0),
    (1, 1, 0),
    (0, 0, 1),
    (1, 0, 1),
    (0, 1, 1),
)

#mendefinisikan sisi
surfaces = (
```

```

#Diamond
(0, 1, 4),
(0, 3, 4),
(2, 1, 4),
(2, 3, 4),
)

#menggambar Limas
def Limastumpuk():
    glBegin(GL_LINES)
    for edge in edges:
        for vertex in edge:
            glVertex3fv(vertices[vertex])
    glEnd()

    glBegin(GL_TRIANGLES)
    for surface in surfaces:
        x = 0
        for vertex in surface:
            x += 1
            glColor3fv(colors[x])
            glVertex3fv(vertices[vertex])
    glEnd()

#menambahkan layar untuk menampilkan limas
def main():
    #inisialisasi pygame
    pygame.init()
    #resolusi display layar
    display = (800,600)
    #mode layar double buffering
    pygame.display.set_mode(display, DOUBLEBUF|OPENGL)
    #mengatur judul pada windows
    pygame.display.set_caption("Menggambar Limas - Mega")
    #mengubah perspektif, fov 45*, znear 0.1, zfar 50
    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)
    #memindahkan objek sesuai dengan matrix translate
    glTranslatef(0.0,0.0, -7)
    #infinite looping
    while True:
        #apabila ditekan tombol x, maka program akan berhenti.
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                pygame.quit()
                quit()
        #apabila ada tombol keyboard yang ditekan,
        #maka akan dilakukan fungsi berikut
        if event.type == pygame.KEYDOWN:

```

```

        #bila yang di tekan tombol panah kiri
        #pindahkan kubus ke kiri sebanyak 0.5
        if event.key == pygame.K_LEFT:
            glTranslatef(-0.5,0,0)
        if event.key == pygame.K_RIGHT:
            glTranslatef(0.5,0,0)

        if event.key == pygame.K_UP:
            glTranslatef(0,1,0)
        if event.key == pygame.K_DOWN:
            glTranslatef(0,-1,0)
    if event.type == pygame.MOUSEBUTTONDOWN:
        if event.button == 4:
            glTranslatef(0,0,1.0)

            if event.button == 5:
                glTranslatef(0,0,-1.0)
    #arah gerak perputaran limas
    glRotatef(1, 1, 1, 0)
    #menghapus semua kanvas/display
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
    Limastumpuk()
    pygame.display.flip()
    #menunggu 10ms sebelum looping lagi
    pygame.time.wait(10)
main()

```


- OUTPUT



