



**PROGRAM STUDI TEKNIK INFORMATIKA**  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN SATYA WACANA

Jl. Dr. O. Notohamidjojo 1-10 Blotongan Salatiga 50715 Jawa Tengah

---

**PORTOFOLIO TUGAS GRAFIKA KOMPUTER**  
**TUGAS 03 ORNAMEN DENGAN BAHASA C DAN OPENGL TOOLS**

---

Nama : Afiyatar Asyer  
NIM : 672019061  
Kelas : IN232 G

Kode Dosen : 67977

---

**1. Screenshot Hasil Program**



## 2. Kode Program dan Penjelasan

Kode Program	Penjelasan Singkat
<p>Diketik rapi dengan font <b>Courier New</b>, 10pt Khusus bagian dasar pemrograman C dicetak warna biru</p> <pre>/* ===== Nama: Afiyatar Asyer NIM : 672019061 ===== */ #include &lt;windows.h&gt; #ifdef __APPLE__ #include &lt;GLUT/glut.h&gt; #else #include &lt;GL/glut.h&gt; #endif #include &lt;stdlib.h&gt; #include &lt;math.h&gt;  void init(void); void display(void);  int main (int argc, char **argv) {     glutInit(&amp;argc, argv);     glutInitDisplayMode(GLUT_DOUBLE   GLUT_RGB);     glutInitWindowSize(640, 480);     glutInitWindowPosition(100, 100);     glutCreateWindow("Tugas3_G_672019061");     init();     glutDisplayFunc(display);     glutMainLoop();     return 0; }  void init(void) {</pre>	<p>Dari program dibutuhkan library include math.h yang nantinya digunakan untuk pembuatan lingkaran atau persegi</p> <p>Penjelasan ornamen:</p> <p>Ornament yang dibuat memiliki nilai filosofi dimana setiap bentukannya memiliki arti antara lain:</p> <ul style="list-style-type: none"><li>• Shape ketupat (merupakan inti dari ornament tersebut yang sangat berharga)</li><li>• Shape segitiga berduri (melambangkan piramida yang penuh dengan jebakan dan mengarah ke shape ketupat melambangkan segitiga ini menyimpan shape ketupat dan terjaga didalamnya)</li><li>• Shape segienam melambangkan sesuatu yang berharga</li><li>• Line melambangkan suatu tekateki/labirin</li></ul>

```

glClearColor(1.0,1.0,1.0,1.0);
glColor3f(0.0,0.0,1.0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-100.0, 100.0, -100.0, 100.0, -100.0, 100.0);
}

```

```

void ngon(int n, float cx, float cy, float radius, float rotAngle){
    double angle, angleInc;
    int k;

    if(n<3) return;
    angle=rotAngle*3.14159265/180;
    angleInc=2*3.14159265/n;

    glVertex2f(radius*cos(angle)+cy, radius*sin(angle)+cy);

    for(k=0;k<n;k++){
        angle+=angleInc;
        glVertex2f(radius*cos(angle)+cy, radius*sin(angle)+cy);
    }
}

```

```

void lingkaran(){
    int segi=0;
    for(int a = 500; a >= 30; a -= 10){
        if(segi%2){
            glBegin(GL_LINE_LOOP);
            glColor3f(1.0,0.0,0.0);
            ngon(6,0,0,a,30);
        }
        else{
            glBegin(GL_LINE_LOOP);
            glColor3f(0.0,0.0,1.0);
            ngon(6,0,0,a,30);
        }
        segi++;
    }
    glEnd();
}

```

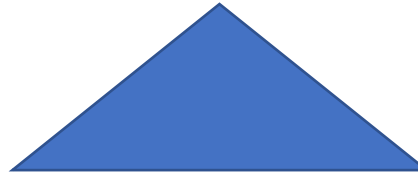
Pembuatan fungsi ngon dan lingkaran dengan menggunakan percabangan dengan kondisi yang telah ditentukan serta terdapat beberapa tipe data yang dimasukkan kedalam parameter

Nantinya fungsi ini akan dipanggil bukan dibuat untuk lingkaran tetapi pembuatan shape dengan segienam karena dari parameter yang diambil nantinya merupakan banyaknya sisi dari shape.



```
}
```

```
void segitiga(float x, float y, float x1, float y1, float x2, float y2){  
    glColor3f(1,1,1);  
    glBegin(GL_TRIANGLES);  
    glColor3f(1.0,0.0,0.5);  
    glVertex2f(x,y);  
    glColor3f(1.0,1.0,0.0);  
    glVertex2f(x1,y1);  
    glColor3f(1.0,1.,0.5);  
    glVertex2f(x2,y2);  
    glEnd();  
}
```



```
void segitiga1(float x, float y, float x1, float y1, float x2, float y2){  
    glColor3f(1,1,1);  
    glBegin(GL_TRIANGLES);  
    glColor3f(1.0,0.0,0.0);  
    glVertex2f(x,y);  
    glColor3f(0.0,0.0,1.0);  
    glVertex2f(x1,y1);  
    glColor3f(1.0,1.0,1.0);  
    glVertex2f(x2,y2);  
    glEnd();  
}
```



```
void segitigaKanan(float x, float y, float x1, float y1, float x2, float y2){
```

```
    for(int a = 0; a < 20; a++){  
        glBegin(GL_TRIANGLES);  
        glColor3f(1,0,0);  
        glVertex2i(x,y);  
        glColor3f(0.0,1.0,0.0);  
        glVertex2i(x1,y1);  
        glColor3f(0.0,0.0,1.0);  
        glVertex2i(x2,y2);  
        x+=5;  
        x1+=5;
```



Fungsi panggil dari segitiga akan diisi nantinya dengan parameter yang telah dibuat dan nantinya setiap vertex akan tersimpan nilainya dengan tipe data float

Dalam pemanggilan fungsi segitiga ada sangat banyak antara lain:

- Segitiga
- Segitiga1
- Segitigakanan
- Segitigakanan1
- Segitigakiri
- Segitigakiril

Semuanya tidak sama antara 1 dan yang lain, bagian pembedanya adalah perwarnaan setiap vertex dengan warna masing-masing sehingga menciptakan sebuah gradasi.

```

x2+=5;
y+=-5;
y1+=-5;
y2+=-5;
glEnd();
}
}

```

```

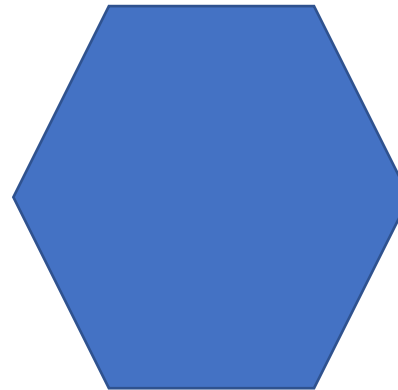
void segienam(float x, float y, float x1, float y1, float x2, float y2, float x3, float y3, float x4, float
y4, float x5, float y5){

```

```

    glBegin(GL_POLYGON);
    glColor3f(1.0,1.0,0.0);
    glVertex2f(x,y);
    glColor3f(1.0,1.0,0.0);
    glVertex2f(x1,y1);
    glColor3f(1.0,0.0,0.0);
    glVertex2f(x2,y2);
    glColor3f(1.0,0.0,1.0);
    glVertex2f(x3,y3);
    glColor3f(1.0,0.0,1.0);
    glVertex2f(x4,y4);
    glColor3f(1.0,0.0,1.0);
    glVertex2f(x5,y5);
    glEnd();
}

```



```

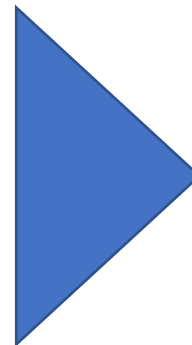
void segitigaKanan1(float x, float y, float x1, float y1, float x2, float y2){

```

```

    for(int a = 0; a < 20; a++){
        glBegin(GL_TRIANGLES);
        glColor3f(1,0,0);
        glVertex2i(x,y);
        glColor3f(0.0,1.0,0.0);
        glVertex2i(x1,y1);
        glColor3f(0.0,0.0,1.0);
        glVertex2i(x2,y2);
        x+=5;
        x1+=5;
        x2+=5;
    }
}

```



Pada bagian shape segi enam dibuat dengan `gl_polygon` dimana pada setiap vertex nantinya akan dibuat sebanyak 6 titik.

Setiap titik akan dibuat warna yang berbeda untuk menciptakan gradasi warna.

Dan untuk mempermudah dibuat fungsi `shape segienam` yang nantinya dipanggil jika dibutuhkan.

Untuk `gl_line_loop` dengan struktur data array akan dibuat sebanyak 4 titik dan 2 sumbu koordinat.

```

y+=5;
y1+=5;
y2+=5;
glEnd();
}
}

```

**void segitigaKiri(float x, float y, float x1, float y1, float x2, float y2){**

```

for(int a = 0; a < 20; a++){
glBegin(GL_TRIANGLES);
glColor3f(1,0,0);
glVertex2i(x,y);
glColor3f(0,0,1);
glVertex2i(x1,y1);
glColor3f(0,0,1);
glVertex2i(x2,y2);
x+=-5;
x1+=-5;
x2+=-5;
y+=5;
y1+=5;
y2+=5;
glEnd();
}
}

```



**void segitigaKiri1(float x, float y, float x1, float y1, float x2, float y2){**

```

for(int a = 0; a < 20; a++){
glBegin(GL_TRIANGLES);
glColor3f(1,0,0);
glVertex2i(x,y);
glColor3f(0,0,1);
glVertex2i(x1,y1);
glColor3f(0,0,1);
glVertex2i(x2,y2);
x+=-5;
x1+=-5;

```



Pada beberapa fungsi segitiga dibuat perulangan agar dapat membuat segitiga yang sama secara berulang.

Contohnya pada bagian:

```

for(int a = 0; a < 20; a++){

glBegin(GL_TRIANGLES);

    glColor3f(1,0,0);

    glVertex2i(x,y);

glColor3f(0,0,1);

    glVertex2i(x1,y1);

glColor3f(0,0,1);

    glVertex2i(x2,y2);

    x+=5;

    x1+=5;

    x2+=5;

    y+=-5;

    y1+=-5;

    y2+=-5;

    glEnd();

}

```

Untuk membuat segitiga duplikatnya

```

x2+=-5;
y+=-5;
y1+=-5;
y2+=-5;
glEnd();
}
}

```

```

void display(void)
{

```

```

    glClearColor(0,0,0,0);

```

```

    int data[4][2]={{-40,-100},{-40,-80},{40,-80},{40,-100}};
    glBegin(GL_LINE_LOOP);
        glColor3f(0.0,0.5,0.5);
        for(int l = 0; l < 4; l++){
            glVertex2i(data[l][0],data[l][1]);
        }
    glEnd();

```

```

    int data1[4][2]={{-40,100},{-40,80},{40,80},{40,100}};
    glBegin(GL_LINE_LOOP);
        glColor3f(0.0,0.5,0.5);
        for(int l = 0; l < 4; l++){
            glVertex2i(data1[l][0],data1[l][1]);
        }
    glEnd();

```

```

    lingkaran();
    segitiga(-20.0,0.0,20.0,0.0,0.0,20.0);
    segitiga(-20.0,0.0,20.0,0.0,0.0,-20.0);
    segitigaKanan(15.0,5.0,15.0,-5.0,20.0,0.0);
    segitigaKanan1(15.0,5.0,15.0,-5.0,20.0,0.0);
    segitiga1(5.0,15.0,-5.0,15.0,0.0,20.0);//
    segitiga1(5.0,-15.0,-5.0,-15.0,0.0,-20.0);//
    segitigaKiri(-15.0,5.0,-15.0,-5.0,-20.0,0.0);

```

Memanggil semua fungsi yang telah dibuat lalu akan dipanggil ke bagian main.

Bagian terakhir adalah dengan memanggil setiap fungsi dan memasukkan nilai kedalam parameternya sebagai patokan nilai. Pemanggilan fungsi antara lain:

- Segitiga
- Segitiga1
- Segitigakanan
- Segitigakanan1
- Segitigakiri
- Segitigakiril
- Segienam
- Lingkaran yang berbentuk segienam dengan banyak sisi 6

```
segitigaKiri1(-15.0,5.0,-15.0,-5.0,-20.0,0.0);  
segienam(0.0,90.0,25.0,75.0,25.0,55.0,0.0,40.0,-25.0,55.0,-25.0,75.0);  
segienam(0.0,-90.0,25.0,-75.0,25.0,-55.0,0.0,-40.0,-25.0,-55.0,-25.0,-75.0);  
segitiga(20.0,0.0,110.0,90.0,110.0,-90.0);  
segitiga(-20.0,0.0,-110.0,90.0,-110.0,-90.0);
```

```
glutSwapBuffers();
```

```
}
```