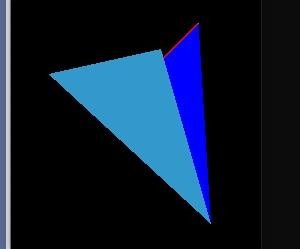
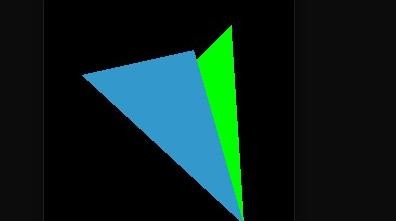
# Nama : Suryandini

# Nim : D0221360

**Kelas : Informatika G**

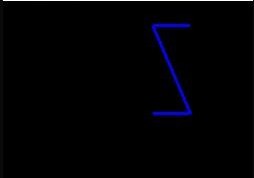
**Latihan1:**



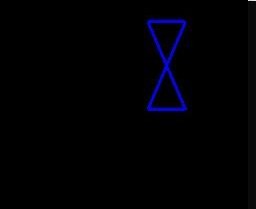


**Latihan2 :**

1. glBegin(GL\_LINE\_STRIP); (Titik awal tidak terhubung dengan titik akhir) glVertex2i(20, 10); glVertex2i(50, 10); glVertex2i(20, 80); glVertex2i(50, 80); glEnd();



1. glBegin(GL\_LINE\_LOOP); (titik awal hingga titik akhir terhubung) glVertex2i(20, 10); glVertex2i(50, 10); glVertex2i(20, 80); glVertex2i(50, 80); glEnd();



1. n-gon simetris/beraturan

glBegin(GL\_LINE\_STRIP);

glVertex2f(40 \* cos(2 \* 3.14159265 \* 1 / 6), 40 \* sin(2 \*

3.14159265 \* 1 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 2 / 6), 40 \* sin(2 \*

3.14159265 \* 2 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 3 / 6), 40 \* sin(2 \*

3.14159265 \* 3 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 4 / 6), 40 \* sin(2 \*

3.14159265 \* 4 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 5 / 6), 40 \* sin(2 \*

3.14159265 \* 5 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 6 / 6), 40 \* sin(2 \*

3.14159265 \* 6 / 6));

glEnd();



1. Tanpa menggunakan Inputan glBegin(GL\_LINE\_LOOP); glVertex2f(40 \* cos(2 \* 3.14159265 \* 1 / 6), 40 \* sin(2 \* 3.14159265

\* 1 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 2 / 6), 40 \* sin(2 \* 3.14159265

\* 2 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 3 / 6), 40 \* sin(2 \* 3.14159265

\* 3 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 4 / 6), 40 \* sin(2 \* 3.14159265

\* 4 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 5 / 6), 40 \* sin(2 \* 3.14159265

\* 5 / 6));

glVertex2f(40 \* cos(2 \* 3.14159265 \* 6 / 6), 40 \* sin(2 \* 3.14159265

\* 6 / 6)); glEnd();



1. Menggunakan Inputan ( fungsi ngon) 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; //titik pertama

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

//titik berikutnya for (k = 0; k < n; k++) { angle += angleInc;

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

} }

void display(void) {

glClear(GL\_COLOR\_BUFFER\_BIT); glBegin(GL\_LINE\_STRIP); ngon(6, 10, 0, 40, 180);

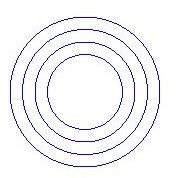
// 6 adalah seginya,40 adlh radiusnya, 180 adlh derajat glEnd();



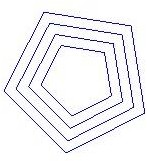
# Latihan 3

**Video 05 :**

* 1. glClear(GL\_COLOR\_BUFFER\_BIT); for (int a = 60; a >= 30; a -= 10) { glBegin(GL\_LINE\_LOOP); ngon(500, 0, 0, a, 45); glEnd(); }



* 1. glClear(GL\_COLOR\_BUFFER\_BIT); for (int a = 60; a >= 30; a -= 10) { glBegin(GL\_LINE\_LOOP); ngon(5, 0, 0, a, 45); glEnd();



* 1. Percabangan segi = 6;

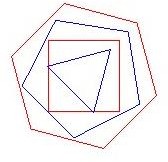
for (int a = 60; a >= 30; a -= 10) { if (segi % 2 == 0) { glBegin(GL\_LINE\_LOOP); glColor3f(1.0, 0.0, 0.0);

ngon(segi, 0, 0, a, 45); glEnd(); } else {

glBegin(GL\_LINE\_LOOP); glColor3f(0.0, 0.0, 1.0);

ngon(segi, 0, 0, a, 45); glEnd();

} segi--;



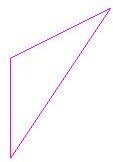
* 1. Struktur Data Array

int data[3][2] = { {0,-40},{0,40},{80,80} }; glBegin(GL\_LINE\_LOOP);

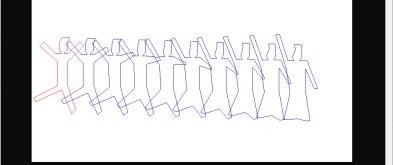
glColor3f(1.0, 0.0, 1.0);

for (int s = 0; s < 3; s++) { glVertex2i(data[s][0], data[s][1]);

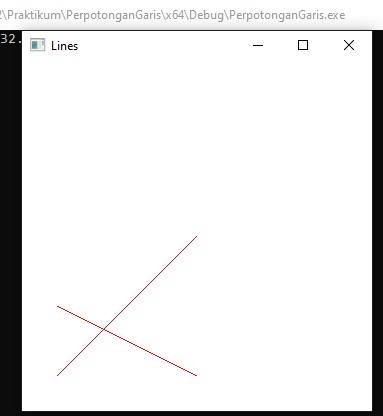
} glEnd();



* 1. Vektor



* 1. Perpotongan Garis



* 1. Menghitung perpotongan garis

