**TECHNICAL REPORT**

**GRAFIKA KOMPUTER**

**MODUL 4**



**Disusun Oleh :**

TGL. PRAKTIKUM : Jum’at, 21 Mei 2021

NAMA : Achmad Farid Alfa Waid

NRP : 190411100073

KELOMPOK : 2

DOSEN : Ach. Khozaimi, S.Kom., M.Kom

TELAH DISETUJUI TANGGAL :

...........................................

ASISTEN PRAKTIKUM

Rizky Alfriansyah

(180411100068)

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

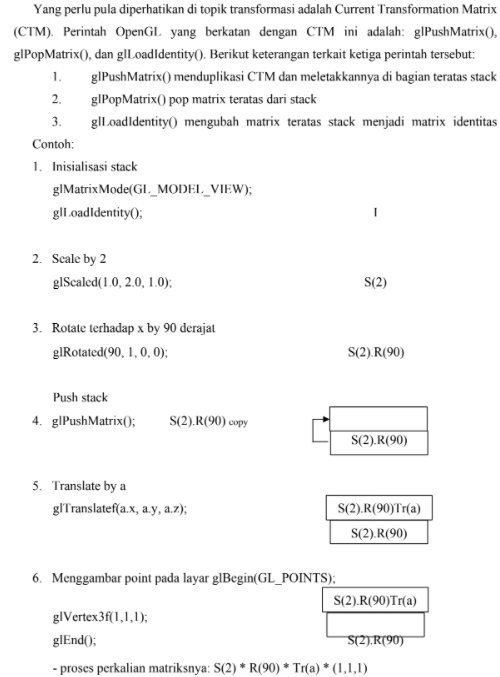
**BAB I**

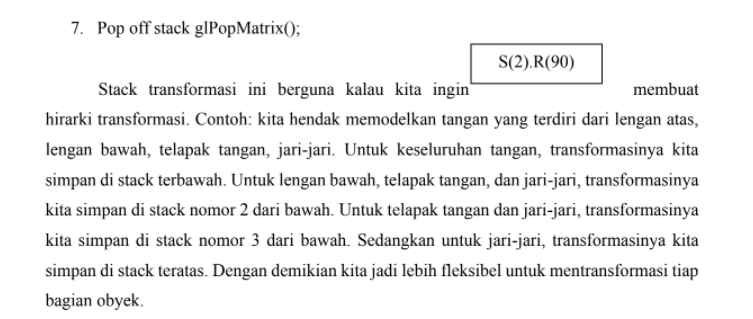
**TUJUAN DAN DASAR TEORI**

1. **TUJUAN**

Mahasiswa dapat mengetahui dan memahami cara melakukan transformasi pada object dengan glPushMatrix dan glPopMatrix.

1. **DASAR TEORI**

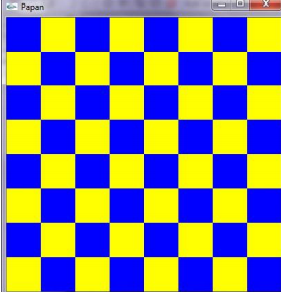
****



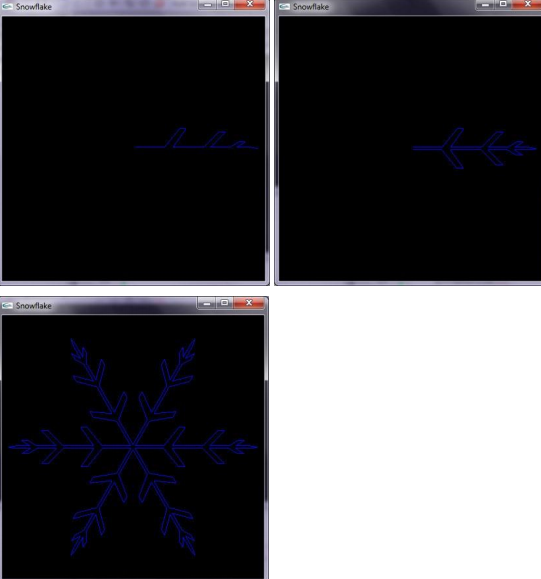
**BAB II**

**PEMBAHASAN**

1. **SOAL**
2. Buat checker board 8 x 8 kotak dengan menggunakan glTranslate.

****

1. Buat snow flake (bunga salju) berikut menggunakan transformasi.

****

Cukup buat 1 bagian, lalu duplikasi menggunakan transformasi 11 kali untuk membuat keseluruhan gambar. Gambar tidak perlu persis, asal cukup mirip.

1. Buat program dengan tampilan sebagai berikut:

****

Gambar tidak harus persis, asal cukup mirip.

1. **JAWABAN**
2. **Code Program**
3. Nomor 1

/\*

\* GLUT Shapes Demo

\*

\* Written by Nigel Stewart November 2003

\*

\* This program is test harness for the sphere, cone

\* and torus shapes in GLUT.

\*

\* Spinning wireframe and smooth shaded shapes are

\* displayed until the ESC or q key is pressed. The

\* number of geometry stacks and slices can be adjusted

\* using the + and - keys.

\*/

#include <windows.h>

#ifdef \_\_APPLE\_\_

#include <GLUT/glut.h>

#else

#include <GL/glut.h>

#endif

#include<iostream>

#include <stdlib.h>

void biru()

{

glColor3f(0,0,1);

glRecti(-20,15,-15,20);

}

void kuning()

{

glColor3f(1,1,0);

glRecti(-15,15,-10,20);

}

void gabunganObject()

{

biru();

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

}

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glPushMatrix();

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-30,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-10,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-20,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-20,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-10,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-30,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,0,0);

gabunganObject();

glPopMatrix();

glFlush ();

}

void myinit()

{

glClearColor(1,1,1,0); //untuk memberi warna pada background

glColor3f(1.0, 0.0, 0.0); // color pada object

glEnable(GL\_POINT\_SMOOTH);

glEnable(GL\_BLEND);

glBlendFunc(GL\_SRC\_ALPHA, GL\_ONE\_MINUS\_SRC\_ALPHA);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-20.0, 20.0, -20.0, 20.0); // kordinat pada area menggambar

glMatrixMode(GL\_MODELVIEW);

}

int main(int argc, char \*argv[])

{

glutInit(&argc,argv);

glutInitWindowSize(400,400);

glutInitWindowPosition(100,100);

glutInitDisplayMode(GLUT\_RGB | GLUT\_SINGLE);

glutCreateWindow("Nomer 1");

glutDisplayFunc(display);

myinit();

glutMainLoop();

return 0;

}

1. Nomor 2

/\*

\* GLUT Shapes Demo

\*

\* Written by Nigel Stewart November 2003

\*

\* This program is test harness for the sphere, cone

\* and torus shapes in GLUT.

\*

\* Spinning wireframe and smooth shaded shapes are

\* displayed until the ESC or q key is pressed. The

\* number of geometry stacks and slices can be adjusted

\* using the + and - keys.

\*/

#include <windows.h>

#ifdef \_\_APPLE\_\_

#include <GLUT/glut.h>

#else

#include <GL/glut.h>

#endif

#include<iostream>

#include <stdlib.h>

void object()

{

glColor3f(0.0, 0.0, 1.0);

glBegin(GL\_LINE\_STRIP);

glVertex3f(0.0, 0.1, 0.0);

glVertex3f(1.75, 0.1, 0.0);

glVertex3f(2.9, 0.8, 0.0);

glVertex3f(3.2, 0.8, 0.0);

glVertex3f(2.3, 0.1, 0.0);

glVertex3f(3.75, 0.1, 0.0);

glVertex3f(4.9, 0.6, 0.0);

glVertex3f(5.2, 0.6, 0.0);

glVertex3f(4.3, 0.1, 0.0);

glVertex3f(5.75, 0.1, 0.0);

glVertex3f(6.6, 0.4, 0.0);

glVertex3f(6.9, 0.4, 0.0);

glVertex3f(6.3, 0.1, 0.0);

glVertex3f(7.5, 0.1, 0.0);

glVertex3f(7.9, 0.0, 0.0);

glEnd();

}

void editObject()

{

glTranslated(0, 0, 0);

object();

glRotatef(180, 1, 0, 0);

glTranslated(0, 0, 0);

object();

glRotatef(180, 0, 1, 0);

glTranslated(0, 0, 0);

object();

glRotatef(180, 1, 0, 0);

glTranslated(0, 0, 0);

object();

}

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glPushMatrix();

glScaled(2, 2, 0);

editObject();

glRotatef(55, 0, 0, 0.5);

glTranslated(0, 0, 0);

editObject();

glRotatef(110, 0, 0, 0.5);

glTranslated(0,0,0);

editObject();

glPopMatrix();

glFlush ();

}

void myinit()

{

glClearColor(1,1,1,0); //untuk memberi warna pada background

glColor3f(1.0, 0.0, 0.0); // color pada object

glEnable(GL\_POINT\_SMOOTH);

glEnable(GL\_BLEND);

glBlendFunc(GL\_SRC\_ALPHA, GL\_ONE\_MINUS\_SRC\_ALPHA);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-20.0, 20.0, -20.0, 20.0); // kordinat pada area menggambar

glMatrixMode(GL\_MODELVIEW);

}

int main(int argc, char \*argv[])

{

glutInit(&argc,argv);

glutInitWindowSize(400,400);

glutInitWindowPosition(100,100);

glutInitDisplayMode(GLUT\_RGB | GLUT\_SINGLE);

glutCreateWindow("Nomer 2");

glutDisplayFunc(display);

myinit();

glutMainLoop();

return 0;

}

1. Nomor 3

/\*

\* GLUT Shapes Demo

\*

\* Written by Nigel Stewart November 2003

\*

\* This program is test harness for the sphere, cone

\* and torus shapes in GLUT.

\*

\* Spinning wireframe and smooth shaded shapes are

\* displayed until the ESC or q key is pressed. The

\* number of geometry stacks and slices can be adjusted

\* using the + and - keys.

\*/

#include <windows.h>

#ifdef \_\_APPLE\_\_

#include <GLUT/glut.h>

#else

#include <GL/glut.h>

#endif

#include<iostream>

#include <stdlib.h>

void object()

{

glColor3f(0.0, 0.0, 1.0);

glBegin(GL\_LINE\_STRIP);

glVertex3f(-0.34, 1.7, 0.0);

glVertex3f(0.13, 1.7, 0.0);

glVertex3f(-0.56, 3.8, 0.0);

glVertex3f(-1.5, 0.81, 0.0);

glVertex3f(-1.16, 0.6, 0.0);

glVertex3f(-0.55, 2.45, 0.0);

glVertex3f(-0.34, 1.7, 0.0);

glEnd();

}

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glScaled(5, 5, 0);

glPushMatrix();

object();

glTranslated(0.35,1,0);

glRotated(70,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-1.07,-0.02,0);

glRotated(290,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-1.46,1,0);

glRotated(220,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-0.55,1.73,0);

glRotated(145,0,0,1);

object();

glPopMatrix();

glFlush ();

}

void myinit()

{

glClearColor(1,1,1,0); //untuk memberi warna pada background

glColor3f(1.0, 0.0, 0.0); // color pada object

glEnable(GL\_POINT\_SMOOTH);

glEnable(GL\_BLEND);

glBlendFunc(GL\_SRC\_ALPHA, GL\_ONE\_MINUS\_SRC\_ALPHA);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-20.0, 20.0, -20.0, 20.0); // kordinat pada area menggambar

glMatrixMode(GL\_MODELVIEW);

}

int main(int argc, char \*argv[])

{

glutInit(&argc,argv);

glutInitWindowSize(400,400);

glutInitWindowPosition(100,100);

glutInitDisplayMode(GLUT\_RGB | GLUT\_SINGLE);

glutCreateWindow("Nomer 3");

glutDisplayFunc(display);

myinit();

glutMainLoop();

return 0;

}

1. **Penjelasan Code Program**
   1. Nomor 1

void biru()

{

glColor3f(0,0,1);

glRecti(-20,15,-15,20);

}

void kuning()

{

glColor3f(1,1,0);

glRecti(-15,15,-10,20);

}

Membuat fungsi kuning dan biru yang didalamnya terdapat syntax glcolor untuk menetukan masing – masing warna sesuai dengan nama fungsi, kemudian membuat sebuah object kotak sekaligus dengan titik koordinatnya dengan fungsi glRecti.

void gabunganObject()

{

biru();

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

glTranslated(10,0,0);

biru();

glTranslated(0,0,0);

kuning();

}

Membuat fungsi gabunganObject yang berisi pemanggilan object biru dan kuning yang sebelumnya sudah dibuat, kemudia melakukan transformasi glTranslated atau pergeseran agar bentuk object yang dibuat sesuai dengan tugas yaitu membentuk papan catur.

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glPushMatrix();

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-30,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-10,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-20,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-20,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,-10,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glTranslated(0,-30,0);

gabunganObject();

glPopMatrix();

glPushMatrix();

glRotated(180,0,0,1);

glTranslated(0,0,0);

gabunganObject();

glPopMatrix();

glFlush ();

}

Kemudian pada fungsi display dipanggil fungsi gabungan object, dimana fungsi tersebut diletakkan didalam glPushMatrix dan glPopMatrix agar fungsi selanjutnya yang akan dipanngil akan kembali ke titik koordinat awal. Hal ini dilakukan sebanyak 8 kali agar terbentuk papan catur seperti pada soal.

1. Nomor 2

void object()

{

glColor3f(0.0, 0.0, 1.0);

glBegin(GL\_LINE\_STRIP);

glVertex3f(0.0, 0.1, 0.0);

glVertex3f(1.75, 0.1, 0.0);

glVertex3f(2.9, 0.8, 0.0);

glVertex3f(3.2, 0.8, 0.0);

glVertex3f(2.3, 0.1, 0.0);

glVertex3f(3.75, 0.1, 0.0);

glVertex3f(4.9, 0.6, 0.0);

glVertex3f(5.2, 0.6, 0.0);

glVertex3f(4.3, 0.1, 0.0);

glVertex3f(5.75, 0.1, 0.0);

glVertex3f(6.6, 0.4, 0.0);

glVertex3f(6.9, 0.4, 0.0);

glVertex3f(6.3, 0.1, 0.0);

glVertex3f(7.5, 0.1, 0.0);

glVertex3f(7.9, 0.0, 0.0);

glEnd();

}

Mmebuat fungsi object untuk membentuk object 1 batang snow dengan menghubungkan masing – masing vertex pada fungsi tersebut. Dan memberikan warna dengan glColor.

void editObject()

{

glTranslated(0, 0, 0);

object();

glRotatef(180, 1, 0, 0);

glTranslated(0, 0, 0);

object();

glRotatef(180, 0, 1, 0);

glTranslated(0, 0, 0);

object();

glRotatef(180, 1, 0, 0);

glTranslated(0, 0, 0);

object();

}

Kemudian membuat sebuah fungsi editObject untuk melekukan transformasi pada object yang ada di fungsi object agar bentuknya sesuai, setelah ditransforamasi menggunakan glRotate dan glTranslate, dipanggil kembali fungsi obect akan membentuk sebuah object gabungan dari beberapa object sebelumnya.

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glPushMatrix();

glScaled(2, 2, 0);

editObject();

glRotatef(55, 0, 0, 0.5);

glTranslated(0, 0, 0);

editObject();

glRotatef(110, 0, 0, 0.5);

glTranslated(0,0,0);

editObject();

glPopMatrix();

glFlush ();

}

Pada fungsi display digunakan glScaled agar object yang terbentuk bertambah besar sebesar 2 pada sumbu x dan y. dan memanggi fungsi editObject agar ditampilkan kemudian dilakukan transformasi menggunakan glRotate dan glTranslate agar bentuk object bergeser dan memiliki kemiringan yang sesuai pada soal dan kemudian dibungkus dengan glPushMatrix dan glPopMatrix.

1. Nomor 3

void object()

{

glColor3f(0.0, 0.0, 1.0);

glBegin(GL\_LINE\_STRIP);

glVertex3f(-0.34, 1.7, 0.0);

glVertex3f(0.13, 1.7, 0.0);

glVertex3f(-0.56, 3.8, 0.0);

glVertex3f(-1.5, 0.81, 0.0);

glVertex3f(-1.16, 0.6, 0.0);

glVertex3f(-0.55, 2.45, 0.0);

glVertex3f(-0.34, 1.7, 0.0);

glEnd();

}

Membuat sebuah fungsi object untuk membuat object awal, yang nantinya akan digabung agar membentuk object baru yang diinginkan dengan menggunakan gabungan dari glVertex agar membentuk object tersebut, dan untuk memberi warna pada object tersebut menggunakan glColor dan object tersebut dibentuk menggunakan fungsi GL\_LINE\_STRIP

void display(void)

{

/\* bersihkan layar dari titik pixel yang masih ada \*/

glClear (GL\_COLOR\_BUFFER\_BIT);

glScaled(5, 5, 0);

glPushMatrix();

object();

glTranslated(0.35,1,0);

glRotated(70,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-1.07,-0.02,0);

glRotated(290,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-1.46,1,0);

glRotated(220,0,0,1);

object();

glPopMatrix();

glPushMatrix();

glTranslated(-0.55,1.73,0);

glRotated(145,0,0,1);

object();

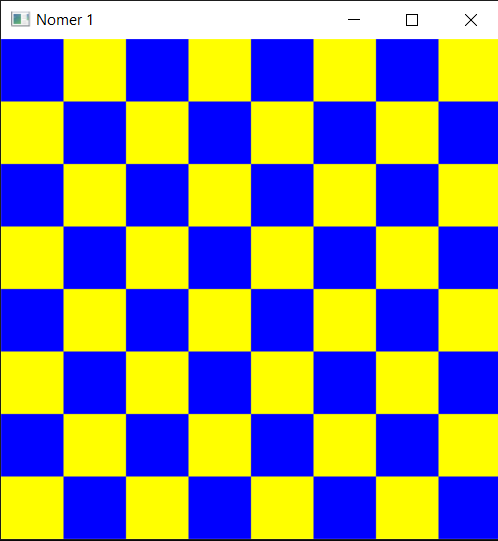
glPopMatrix();

glFlush ();

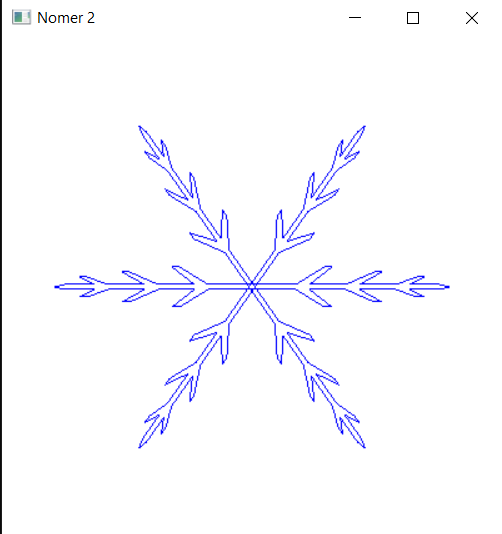
}

Kemudian pada fungsi display melakukan transformasi glScaled sebesar 5 pada sumbu x dan y, dan transformasi glTranslated agar object yang diasilkan bergeser ke tengah, selanjutnya memanggil fungsi Object dan diletakkan diantara glPushMatrix dan glPopMatrix agar fungsi selanjutnya ayang akan dipanngil kembali ke titik koordinat awal, dan akhirnya object yang digambar atau dibuat tampil sesuai output pada soal.

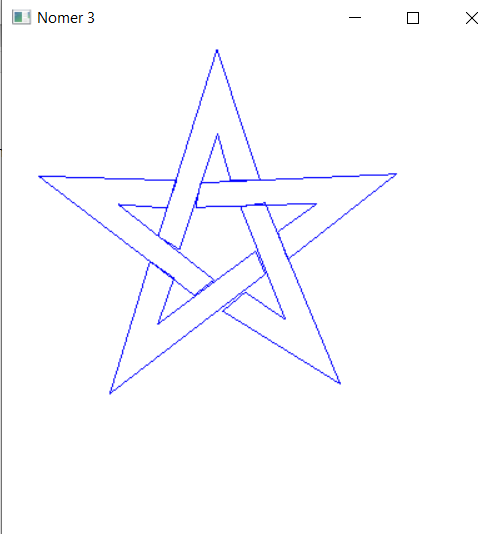
1. **Hasil Running Program** 
   1. Nomor 1



1. Nomor 2



1. Nomor 3



**BAB III**

**PENUTUP**

1. **Kesimpulan**
2. Mahasiswa mampu melakukan transformasi pada object yang dibuat menggunakan glPushMatrix dan glPopMatrix.
3. Terdapat 3 fungsi transformasi yang dapat digunakan, dan masing – masing transformasi dapat digabungkan dalam pemakaiannya.
4. glPushMatrix dan glPopMatrix sangatpenting untuk digunakan.
5. **Saran**

Diperlukan banyak latihan dan mencoba megotak-atik sendiri agar mahasiswa lebih memahami dan mendalami fungsi – fungsi yang ada pada openGL.