

LAPORAN PRAKTIKUM
GRAFIKA DAN KOMPUTASI VISUAL
CYBERTRUCK



Dibuat Untuk Memenuhi Tugas Besar Pembuatan Animasi Truk

Disusun Oleh :

- | | |
|------------------------------|------------------|
| 1. Firdaus Ravi Faizzani | (24060121130067) |
| 2. Muhammad Hilmy Tsany | (24060120140171) |
| 3. Rizal Zeri Subakti | (24060120130062) |
| 4. Victorio Cristiansa Putra | (24060121130084) |

PROGRAM STUDI S-1 INFORMATIKA
FAKULTAS SAINS DAN MATEMATIKA
UNIVERSITAS DIPONEGORO
SEMARANG

2022

DAFTAR ISI

Daftar Isi.....	1
BAB I.....	2
1.1 Latar Belakang.....	2
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
BAB 2.....	3
2.1 OpenGL.....	3
2.2 Lighting OpenGL.....	4
2.3 Windowing OpenGL.....	4
BAB 3.....	5
3.1 Truk.....	5
3.2 Pohon.....	14
3.3 Gedung.....	20
3.4 Lantai.....	22
3.5 Jalan dan Tembok.....	24
3.6 Istouch dan Game Over.....	28
3.7 Object.....	29
BAB 4	31
4.1 Kesimpulan.....	31
Daftar Pustaka.....	32
Lampiran.....	33

BAB I

PENDAHULUAN

1.1 Latar Belakang

Digital Image Processing, Komputer Grafik, Analisis dan Komputer telah dikembangkan dan diaplikasikan dengan mengesankan selama beberapa dekade. Dimana perkembangan aplikasi-aplikasi yang menggunakan disiplin ilmu telah memimpin teknologi di beberapa bidang seperti komunikasi digital dan internet, penyiaran (broadcasting), alat kedokteran, sistem multimedia, biologi, ilmu pengetahuan material, robot dan manufaktur, sistem intelligent sensing, remote sensing, seni grafik dan proses print. Pertumbuhan yang pesat ini direfleksikan dengan diterbitkannya paper di jurnal ilmiah internasional setiap tahunnya dan diluncurkannya buku-buku tentang Pemrosesan Image Digital dan Komputer Grafik.

Dalam membuat sebuah animasi dibutuhkan OpenGL. OpenGL (Open Graphics Library) adalah standar API yang dapat digunakan untuk membuat aplikasi berbasis grafik, baik dua dimensi (2D) maupun tiga dimensi (3D). OpenGL ini bersifat cross-platform, artinya dapat dijalankan pada berbagai platform sistem operasi yang ada saat ini. Dalam mengoperasikan OpenGL, dapat ditambahkan GLUT yang merupakan sebuah library pada OpenGL dan berfungsi untuk mempermudah dalam penggunaan OpenGL.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, didapatkan rumusan masalah sebagai berikut.

1. Bagaimana proses pembuatan Truk dengan menggunakan OpenGL dan pengaplikasiannya
2. Bagaimana hasil eksekusi program saat dijalankan?

1.3 Tujuan

Berdasarkan rumusan masalah, tujuan dari pembuatan laporan ini sebagai berikut.

- a. Dapat memahami proses pembuatan Truk dengan menggunakan OpenGL dan pengaplikasiannya
- b. Dapat menampilkan hasil setelah program dijalankan.

BAB II

DASAR TEORI

2.1 OpenGL

OpenGL (Open Graphic Library) merupakan library yang terdiri dari berbagai macam fungsi dan biasanya digunakan untuk menggambar sebuah atau beberapa objek 2 dimensi dan 3 dimensi. Library-library ini mendefinisikan sebuah cross-bahasa, cross-platform API (antarmuka pemrograman aplikasi) untuk menulis aplikasi yang menghasilkan komputer 2D dan 3D grafis. Bahasa pemrograman yang digunakan pada umumnya adalah pemrograman C/C++, namun dapat pula menggunakan Java. OpenGL merupakan library yang digunakan untuk melakukan pemrograman grafik; Graphic Programming.

Secara garis besar, OpenGL memiliki beberapa tahapan operasi pemrosesan. Tahap pertama adalah display list, yaitu pengguna bisa mengumpulkan beberapa perintah-perintah yang digunakan pada display list untuk diproses pada nantinya. Tahap kedua adalah evaluator, yang melakukan evaluasi perintah polinomial nilai-nilai input untuk melakukan perhitungan pada kurva dan geometri permukaan. Tahap ketiga adalah per-vertex operations and primitive assembly, yaitu tahapan pemrosesan pada OpenGL di mana sekumpulan vertex ditransformasikan dan primitives dipotong dan dimasukkan ke dalam viewport untuk dilakukan proses rasterization. Tahap keempat adalah rasterization, di mana sekumpulan alamat frame-buffer dan nilai-nilai yang berkaitan dihasilkan dengan deskripsi dua dimensi baik mengenai titik, segmen garis, maupun poligon. Tahap terakhir adalah per-fragment operations, yang melibatkan perbaharuan tertentu pada frame-buffer berdasarkan nilai z yang telah disimpan dan yang akan dimasukkan, di mana nilai z ini berguna untuk z-buffering

2.2 Lighting OpenGL

Pencahayaan pada OpenGL hanya dengan cahaya pendekatan dan cahaya lampu seolah-olah dapat dipecah menjadi komponen merah, hijau, dan biru. Dengan demikian, warna sumber cahaya dicirikan oleh jumlah warna yang memancarkan cahaya merah, hijau, dan biru, dan materi permukaan ditandai dengan persentase komponen warna merah, hijau, dan biru yang masuk dan tercermin dalam berbagai arah. Persamaan pencahayaan pada OpenGL hanya sebuah pendekatan, tetapi satu yang bekerja cukup baik dan dapat dihitung relatif cepat. Jika menginginkan yang lebih akurat atau model pencahayaan hanya berbeda, harus melakukan perhitungan sendiri dalam perangkat lunak. Dengan perangkat lunak tersebut dapat menjadi sangat kompleks, seperti beberapa jam membaca buku teks optik pun harus meyakinkan.

2.3 Windowing OpenGL

Ada 5 fungsi yang harus dipanggil untuk melakukan instalansi window yaitu:

a. `glutInit(int argc, char**argv)`

Fungsi ini menginisialisasi glut dan memproses argument command_line yg disertakan(jika ada). Fungsi ini harus di panggil pertama kali sebelum pemanggilan terhadap fungsi-fungsi yang lain.

b. `glutInitDisplayMode(unsigned int mode)`

Fungsi ini digunakan untuk menentukan apakah akan menggunakan model pewarnaan RGBA atau indeks warna. Dengan fungsi ini juga bisa menentukan apakah akan menggunakan windows buffer single atau double.

c. `GlutInitWindowPosission(int x, int y)`

Fungsi ini menentukan letak window dengan patokkannya ialah pada window kiri atas.

d. `glutInitWindowSize(int width, int height)`

Fungsi ini menspesifikasikan ukuran window yang dibuat dalam pixel.

e. `glutCreateWindow(char *string)`

Fungsi untuk membuat window dalam konteks openGL, dan akan menghasilkan identitas atau judul window tersebut.

BAB III

PEMBAHASAN

3.1 Truk

```
void Roda(float x_roda, float y_roda, float z_roda) {
    glPushMatrix();
    glRotated(90, 0, 0, 1);
    glTranslatef(x_roda,y_roda,z_roda);
    glScalef(3.,3.,3.);
    glColor3ub(40, 40,40);
    // Kanan Kiri Roda
    glPushMatrix();
        glBegin(GL_TRIANGLE_FAN);
        for (int a = 0; a <= 360; a+=1){
            glVertex3f(cos(a), 1, sin(a));
        }
        glEnd();
    glPopMatrix();

    glPushMatrix();
        glTranslatef(0.0, 1.0, 0.0);
        glBegin(GL_TRIANGLE_FAN);
        for (int a = 0; a <= 360; a+=1){
            glVertex3f(cos(a), 1, sin(a));
        }
        glEnd();
    glPopMatrix();

    // Tengah Roda
    glPushMatrix();
        int b;
        glColor3ub(10, 10, 10);
        glTranslatef(0.0, 1.0, 0.0);
        glBegin(GL_QUAD_STRIP);
        for (b=0; b<=360; b+=1){
            glVertex3f(cos(b), +1, sin(b));
            glVertex3f(cos(b), 0, sin(b));
        }
        glEnd();
    glPopMatrix();
    glPopMatrix();
}

void TrukDepan(){
    /////////////// Sisi Kiri ///////////////////
    glPushMatrix();
    glScalef(2,2,2);
    glRotated(-90,0,1,0);
    glTranslatef(-10,-0.7,2);
    glColor3ub(20,20,20);
    glBegin(GL_TRIANGLE_STRIP);
    // Kaca
    glVertex2f(4,4.3);
```

```

glVertex2f(4,3.7);
glVertex2f(6,4.6);
glVertex2f(8,3.5);
glVertex2f(8,4.7);
glVertex2f(12.1,3.3);
glEnd();
// Body
    glColor3ub(100,100,100);
    glBegin(GL_QUADS);
    // Bawah Belakang
    glVertex2f(0,4);
    glVertex2f(0.3,2.4);
    glVertex2f(4,2.3);
    glVertex2f(4,3.7);
    // Bawah Tengah
    glVertex2f(4,2.3);
    glVertex2f(4,3.7);
    glVertex2f(12.1,3.3);
    glVertex2f(12,2.2);
    // Bawah Depan
    glVertex2f(12.1,3.3);
    glVertex2f(12,2.2);
    glVertex2f(13.8,2.2);
    glVertex2f(14,3);
    // Depan Atas
    glVertex2f(12.1,3.3);
    glVertex2f(14,3);
    glVertex2f(8,5);
    glVertex2f(8,4.7);
    // Atas Tengah
    glVertex2f(8,5);
    glVertex2f(8,4.7);
    glVertex2f(6,4.6);
    glVertex2f(6,5);
    // Belakang Tengah
    glVertex2f(6,4.6);
    glVertex2f(6,5);
    glVertex2f(4,4.7);
    glVertex2f(4,4.3);
    // Belakang Atas
    glVertex2f(4,4.7);
    glVertex2f(4,4.3);
    glVertex2f(4,3.7);
    glVertex2f(0,4);
    // Outline Bawah
    glColor3ub(15,15,15);
    glVertex2f(0.3,2.4);
    glVertex2f(0.5,1.4);
    glVertex2f(13.6,1.4);
    glVertex2f(13.8,2.2);
    glEnd();
glPopMatrix();

////////// Sisi Kanan //////////
glPushMatrix();
glScalef(2,2,2);

```

```

glRotated(-90,0,1,0);
glTranslatef(-10,-0.7,-2);
glColor3ub(20,20,20);
glBegin(GL_TRIANGLE_STRIP);
// Kaca
glVertex2f(4,4.3);
glVertex2f(4,3.7);
glVertex2f(6,4.6);
glVertex2f(8,3.5);
glVertex2f(8,4.7);
glVertex2f(12.1,3.3);
glEnd();
// Body
    glColor3ub(100,100,100);
    glBegin(GL_QUADS);
    // Bawah Belakang
    glVertex2f(0,4);
    glVertex2f(0.3,2.4);
    glVertex2f(4,2.3);
    glVertex2f(4,3.7);
    // Bawah Tengah
    glVertex2f(4,2.3);
    glVertex2f(4,3.7);
    glVertex2f(12.1,3.3);
    glVertex2f(12,2.2);
    // Bawah Depan
    glVertex2f(12.1,3.3);
    glVertex2f(12,2.2);
    glVertex2f(13.8,2.2);
    glVertex2f(14,3);
    // Depan Atas
    glVertex2f(12.1,3.3);
    glVertex2f(14,3);
    glVertex2f(8,5);
    glVertex2f(8,4.7);
    // Atas Tengah
    glVertex2f(8,5);
    glVertex2f(8,4.7);
    glVertex2f(6,4.6);
    glVertex2f(6,5);
    // Belakang Tengah
    glVertex2f(6,4.6);
    glVertex2f(6,5);
    glVertex2f(4,4.7);
    glVertex2f(4,4.3);
    // Belakang Atas
    glVertex2f(4,4.7);
    glVertex2f(4,4.3);
    glVertex2f(4,3.7);
    glVertex2f(0,4);
    // Outline Bawah
    glColor3ub(15,15,15);
    glVertex2f(0.3,2.4);
    glVertex2f(0.5,1.4);
    glVertex2f(13.6,1.4);
    glVertex2f(13.8,2.2);

```



```

        glEnd();
glPopMatrix();
////////// PENUTUP //////////
glPushMatrix();
    glTranslatef(4,-1.4,-20);
    glRotated(-90,0,1,0);
    glScalef(2,2,2);
    glColor3ub(110,110,110);
    glBegin(GL_QUADS);
        // Depan
        glVertex3f(13.8,2.2,0);
        glVertex3f(13.8,2.2,4);
        glVertex3f(14,3,4);
        glVertex3f(14,3,0);
        // Depan Atas
        glVertex3f(14,3,4);
        glVertex3f(14,3,0);
        glVertex3f(13,3.3,0);
        glVertex3f(13,3.3,4);
        // Belakang Atas
        glVertex3f(6,5,0);
        glVertex3f(6,5,4);
        glVertex3f(0,4,4);
        glVertex3f(0,4,0);
        // Bawah
        glVertex3f(0.5,1.4,0);
        glVertex3f(0.5,1.4,4);
        glVertex3f(13.6,1.4,4);
        glVertex3f(13.6,1.4,0);
        // Kaca Depan
        glColor3ub(20,20,20);
        glVertex3f(13,3.3,0);
        glVertex3f(13,3.3,4);
        glVertex3f(8,5,4);
        glVertex3f(8,5,0);
        // Bumper Depan
        glColor3ub(5,5,5);
        glVertex3f(13.6,1.4,0);
        glVertex3f(13.6,1.4,4);
        glVertex3f(13.8,2.2,4);
        glVertex3f(13.8,2.2,0);
        // Lampu Belakang
        glColor3ub(175,69,42);
        glVertex3f(0.3,2.4,0);
        glVertex3f(0.3,2.4,4);
        glVertex3f(0.5,1.4,4);
        glVertex3f(0.5,1.4,0);
        // Belakang
        glColor3ub(1,133,121);
        glVertex3f(0,4,4);
        glVertex3f(0,4,0);
        glVertex3f(0.3,2.4,0);
        glVertex3f(0.3,2.4,4);
        // Atap
        glVertex3f(8,5,4);
        glVertex3f(8,5,0);

```

```

        glVertex3f(6,5,0);
        glVertex3f(6,5,4);
    glEnd();
    glPopMatrix();
    //////////////////////////////////Roda////////////////////////////////////
    Roda(3.2,1,1.9);
    Roda(3.2,-10,1.9);
    Roda(3.2,1,-15);
    Roda(3.2,-10,-15);
    ////////////////////////////////// Pengait //////////////////////////////////
    glPushMatrix();
    glColor3f(1,1,1);
    glScalef(2,2,2);
    glRotated(90,0,1,0);
    glTranslatef(7.5,0,-0.5);
    glBegin(GL_QUADS);
        glVertex3f(1,1,0);
        glVertex3f(1,1,1);
        glVertex3f(1,2,1);
        glVertex3f(1,2,0);

        glVertex3f(1,2,1);
        glVertex3f(1,2,0);
        glVertex3f(7,2,0);
        glVertex3f(7,2,1);

        glVertex3f(7,2,0);
        glVertex3f(7,2,1);
        glVertex3f(7,1,1);
        glVertex3f(7,1,0);

        glVertex3f(7,1,1);
        glVertex3f(7,1,0);
        glVertex3f(1,1,0);
        glVertex3f(1,1,1);

        glVertex3f(7,2,0);
        glVertex3f(7,1,0);
        glVertex3f(1,1,0);
        glVertex3f(1,2,0);

        glVertex3f(1,1,1);
        glVertex3f(1,2,1);
        glVertex3f(7,2,1);
        glVertex3f(7,1,1);
    glEnd();
    glPopMatrix();
}

void TrukBelakang()
{
    // Kiri
    glPushMatrix();
    glTranslatef(-5,0,-23);
    glScalef(2,2,2);
    glRotated(-90,0,1,0);

```

```

glColor3ub(35,35,35);
glBegin(GL_TRIANGLE_STRIP);
glVertex3f(-1.3,3.,0.); // Titik Tengah Depan
glVertex3f(-4.5,7.,0.); // titik Atas Depan
glVertex3f(-2.5,2.,0.); // Titik Bawah Depan
glVertex3f(-14.,7.,0.); // Titik Atas Belakang
glVertex3f(-12.,2.,0.); // Titik Bawah Tengah
glVertex3f(-14.,2.,0.); // Titik Bawah Belakang
glEnd();
// Outline Kiri
glBegin(GL_QUADS);
glColor3ub(155,155,155);
    // Bawah
    glVertex3f(-15,1,0);
    glVertex3f(-15,2,0);
    glVertex3f(-1,2,0);
    glVertex3f(-2,1,0);
    // Depan Bawah
    glVertex3f(-2.5,2,0);
    glVertex3f(-1,2,0);
    glVertex3f(0,3,0);
    glVertex3f(-1.3,3,0);
    // Depan Atas
    glVertex3f(0,3,0);
    glVertex3f(-1.3,3,0);
    glVertex3f(-4.5,7,0);
    glVertex3f(-4,8,0);
    // Atas
    glVertex3f(-4.5,7,0);
    glVertex3f(-4,8,0);
    glVertex3f(-15,8,0);
    glVertex3f(-14,7,0);
    // Belakang
    glVertex3f(-15,8,0);
    glVertex3f(-14,7,0);
    glVertex3f(-14,2,0);
    glVertex3f(-15,2,0);
glEnd();
glPopMatrix();

// Kanan
glPushMatrix();
glTranslatef(5,0,-23);
glScalef(2,2,2);
glRotated(-90,0,1,0);
glColor3ub(35,35,35);
glBegin(GL_TRIANGLE_STRIP);
glVertex3f(-1.3,3.,0.); // Titik Tengah Depan
glVertex3f(-4.5,7.,0.); // titik Atas Depan
glVertex3f(-2.5,2.,0.); // Titik Bawah Depan
glVertex3f(-14.,7.,0.); // Titik Atas Belakang
glVertex3f(-12.,2.,0.); // Titik Bawah Tengah
glVertex3f(-14.,2.,0.); // Titik Bawah Belakang
glEnd();
// Outline Kanan
glBegin(GL_QUADS);

```

```

glColor3ub(155,155,155);
// Bawah
glVertex3f(-15,1,0);
glVertex3f(-15,2,0);
glVertex3f(-1,2,0);
glVertex3f(-2,1,0);
// Depan Bawah
glVertex3f(-2.5,2,0);
glVertex3f(-1,2,0);
glVertex3f(0,3,0);
glVertex3f(-1.3,3,0);
// Depan Atas
glVertex3f(0,3,0);
glVertex3f(-1.3,3,0);
glVertex3f(-4.5,7,0);
glVertex3f(-4,8,0);
// Atas
glVertex3f(-4.5,7,0);
glVertex3f(-4,8,0);
glVertex3f(-15,8,0);
glVertex3f(-14,7,0);
// Belakang
glVertex3f(-15,8,0);
glVertex3f(-14,7,0);
glVertex3f(-14,2,0);
glVertex3f(-15,2,0);
glEnd();
glPopMatrix();

////////// Penutup //////////
glPushMatrix();
glTranslatef(-5,0,-23);
glScalef(2,2,2);
glBegin(GL_QUADS);
// Depan Bawah
glVertex3f(0.,3.,0.);
glVertex3f(5.,3.,0.);
glVertex3f(5.,1.,-2.);
glVertex3f(0.,1.,-2.);
// Depan Atas
glColor3ub(35,35,35);
glVertex3f(0.,3.,0.);
glVertex3f(0.,8.,-4.);
glVertex3f(5.,8.,-4.);
glVertex3f(5.,3.,0.);
// Atas
glVertex3f(0.,8.,-4.);
glVertex3f(5.,8.,-4.);
glVertex3f(5.,8.,-15.);
glVertex3f(0.,8.,-15.);
// Belakang
glColor3ub(15,15,15);
glVertex3f(0.,7.,-15.);
glVertex3f(5.,7.,-15.);
glVertex3f(5.,2.,-15.);
glVertex3f(0.,2.,-15.);

```

```

// Outline Belakang
glColor3ub(170,170,0);
glVertex3f(0.,7.,-15.);
glVertex3f(5.,7.,-15.);
glVertex3f(5.,8.,-15.);
glVertex3f(0.,8.,-15.);

glVertex3f(5.,2.,-15.);
glVertex3f(0.,2.,-15.);
glVertex3f(0.,1.,-15.);
glVertex3f(5.,1.,-15.);
// Bawah
glColor3ub(120,70,140);
glVertex3f(0.,1.,-15.);
glVertex3f(5.,1.,-15.);
glVertex3f(5.,1.,-2.);
glVertex3f(0.,1.,-2.);
glEnd();
glPopMatrix();
////////// Roda //////////
Roda(3.2,2,-48);
Roda(3.2,-11,-48);
Roda(3.2,2,-31);
Roda(3.2,-11,-31);
}

```

Di atas merupakan source code dari bagian Truk, truk tersebut terdiri dari beberapa bagian yang membentuk satu kesatuan truk, bagian-bagian tersebut dipisah menjadi beberapa procedure yaitu: roda, trukdepan, dan trukbelakang.

Procedure pertama yaitu bagian roda truk depan dan belakang bernama void Roda() yang memiliki 3 parameter yaitu x, y, z dimana ketiga tipe data parameter tersebut berupa float. Bentuk roda tersebut menggunakan GL_TRIANGLE_FAN yang di looping, kemudian untuk bagian tengah roda menggunakan GL_QUAD_STRIP yang di looping juga.

Procedure kedua yaitu bagian truk depan bernama void TrukDepan() yang tidak memiliki parameter. Di bagian sisi kirinya menggunakan GL_TRIANGLE_STRIP, lalu terdapat bagian kaca dan bagian body yang menggunakan GL_QUADS, untuk bagian sisi kirinya juga menggunakan GL_TRIANGLE_STRIP dan terdapat bagian kaca dan body nya. Kemudian, terdapat bagian penutup bagian atas dari truk depan serta roda dan pengait antara truk depan dan belakang sehingga membentuk cybertruck.

Procedure ketiga yaitu bagian truk belakang bernama void TrukBelakang(), procedure ini mirip seperti procedure sebelumnya yang membentuk truk, namun pada procedure ini akan membentuk truk yang di derek dari truk depannya.

```

float posXTruk = 0.0, posZTruk = 0.0;

void TrukMove()
{
    //Belok kanan
}

```

```

    if (belok == -1){
        if (angTrukDepan >= -10){
            angTrukDepan -= 2;
        }
        if (angTrukDepan < -9.9){
            belok = -2;
        }
    }
    if (belok <= -2){
        if (angTrukDepan <= -0.5){
            angTrukDepan += 2;
            posXTruk -= 1;
        }
        if (angTrukDepan > -0.1){
            belok = 0;
        }
    }
}

//Belok kiri
if (belok == 1){
    if (angTrukDepan <= 10){
        angTrukDepan += 2;
    }
    if (angTrukDepan > 9.9){
        belok = 2;
    }
}
if (belok >= 2){
    if (angTrukDepan >= 0.5){
        angTrukDepan -= 2;
        posXTruk += 1;
    }
    if (angTrukDepan < 0.1){
        belok = 0;
    }
}
}

float angTrukBelakang = 0;

void TrukMove2()
{
    if (belok == -1 and angTrukDepan <= -4.9){
        if (angTrukBelakang >= -10){
            angTrukBelakang -= 2;
        }
    }
    if (belok == -2 and angTrukDepan >= -5.1){
        if (angTrukBelakang <= -0.5){
            angTrukBelakang += 2;
        }
    }
    if (belok == 1 and angTrukDepan >= 4.9){
        if (angTrukBelakang <= 10){
            angTrukBelakang += 2;
        }
    }
}

```

```

    }
    if (belok == 2 and angTrukDepan <= 5.1){
        if (angTrukBelakang >= 0.5){
            angTrukBelakang -= 2;
        }
    }
}

float speed = 0;
float posZJalan = -1000;
int ujung = 0;

```

Lalu untuk pergerakan truk menggunakan procedure TrukMove(), dan TrukMove2(), pada procedure ini menggunakan banyak conditional if.

3.2 Pohon

```

// Fungsi untuk membuat pohon jenis 1
void Pohon1( int r, int g, int b) {
    glPushMatrix();
        glRotatef(0,0,1,0);
        glTranslatef(-40,0,0);
        glScalef(3,3,3);
        glBegin(GL_QUADS);
        // Batang Pohon
        glColor3ub(248, 144, 32);
        glVertex2f(4,0);
        glVertex2f(3.6,0);
        glVertex2f(3.2,1.5);
        glVertex2f(3.6,1.3);

        glVertex2f(4,0);
        glVertex2f(4.6,0);
        glVertex2f(3.9,1.4);
        glVertex2f(3.6,1.3);

        glVertex2f(3.6,1.3);
        glVertex2f(3.2,1.5);
        glVertex2f(3.4,3.2);
        glVertex2f(3.6,3);

        glVertex2f(3.6,1.3);
        glVertex2f(3.9,1.4);
        glVertex2f(3.9,3.1);
        glVertex2f(3.6,3);

        glVertex2f(3.6,3);
        glVertex2f(3.4,3.2);
        glVertex2f(3.9,4.3);
        glVertex2f(4.1,4.4);

        glVertex2f(3.6,3);
        glVertex2f(3.9,3.1);
        glVertex2f(4.2,4.3);

```

```

glVertex2f(4.1,4.4);

// Daun Kanan Bawah
glColor3ub(r, g, b);
glVertex2f(4.1,4.4);
glVertex2f(4.5,4.6);
glVertex2f(4.7,4.4);
glVertex2f(4.5,4.2);

glVertex2f(4.2,4.3);
glVertex2f(4.5,4.6);
glVertex2f(4.7,4.4);
glVertex2f(4.5,4.2);

glVertex2f(4.1,4.4);
glVertex2f(4.5,4.6);
glVertex2f(4.7,4.4);
glVertex2f(4.2,4.3);

glVertex2f(4.5,4.2);
glVertex2f(4.7,4.4);
glVertex2f(5.2,3.7);
glVertex2f(5,3.5);

glVertex2f(5.1,4.3);
glVertex2f(4.7,4.4);
glVertex2f(5.2,3.7);
glVertex2f(5.4,3.6);

glVertex2f(5.1,4.3);
glVertex2f(4.7,4.4);
glVertex2f(4.5,4.6);
glVertex2f(4.8,4.75);

glVertex2f(5,3.5);
glVertex2f(5.2,3.7);
glVertex2f(5.4,3.6);
glVertex2f(5.2,3);

///// Daun Kiri Bawah
glVertex2f(3.71,4.88);
glVertex2f(3.92,4.78);
glVertex2f(4.1,4.71);
glVertex2f(3.27,4.32);

glVertex2f(3.32,4.15);
glVertex2f(4.11,4.42);
glVertex2f(4.1,4.71);
glVertex2f(3.27,4.32);

glVertex2f(3.32,4.15);
glVertex2f(4.11,4.42);
glVertex2f(3.92,4.26);
glVertex2f(2.88,3.54);

glVertex2f(3.32,4.15);

```



```

glVertex2f(2.72,3.65);
glVertex2f(2.89,4.14);
glVertex2f(3.27,4.32);

glVertex2f(3.32,4.15);
glVertex2f(2.72,3.65);
glVertex2f(2.47,3.59);
glVertex2f(2.89,4.14);

glVertex2f(3.32,4.15);
glVertex2f(2.88,3.54);
glVertex2f(2.72,3.65);
glVertex2f(2.89,4.14);

glVertex2f(2.88,3.54);
glVertex2f(2.72,3.65);
glVertex2f(2.47,3.59);
glVertex2f(2.4,2.8);

// Daun Kanan Tengah
glVertex2f(4.1,4.4);
glVertex2f(4.57,4.96);
glVertex2f(5.18,4.76);
glVertex2f(4.8,4.65);

glVertex2f(4.64,5.03);
glVertex2f(4.57,4.96);
glVertex2f(5.18,4.76);
glVertex2f(5.1,5.02);

glVertex2f(6.23,5.02);
glVertex2f(6.17,4.7);
glVertex2f(5.18,4.76);
glVertex2f(5.1,5.02);

glVertex2f(6.23,5.02);
glVertex2f(6.75,4.97);
glVertex2f(5.83,5.5);
glVertex2f(5.1,5.02);

glVertex2f(4.64,5.03);
glVertex2f(4.83,5.27);
glVertex2f(5.83,5.5);
glVertex2f(5.1,5.02);

glVertex2f(6.23,5.02);
glVertex2f(6.17,4.7);
glVertex2f(7.12,4.34);
glVertex2f(6.75,4.97);

//// Daun Kiri Tengah
glVertex2f(3.71,4.88);
glVertex2f(2.49,4.67);
glVertex2f(2.14,4.9);
glVertex2f(2.84,4.94);

```

```

glVertex2f(3.71,4.88);
glVertex2f(2.49,4.67);
glVertex2f(2.3,4.39);
glVertex2f(3.27,4.32);

glVertex2f(1.46,4.27);
glVertex2f(2.49,4.67);
glVertex2f(2.3,4.39);
glVertex2f(1.48,3.99);

glVertex2f(1.46,4.27);
glVertex2f(2.49,4.67);
glVertex2f(2.14,4.9);
glVertex2f(1.34,4.35);

glVertex2f(1.46,4.27);
glVertex2f(1.48,3.99);
glVertex2f(1.05,3.71);
glVertex2f(1.34,4.35);

//// Daun Kanan ATAS
glColor3ub(r-21, g-22, b-23);
glVertex2f(4.1,4.4);
glVertex2f(4.1,4.71);
glVertex2f(4.34,5.3);
glVertex2f(4.83,5.27);

glVertex2f(4.57,4.96);
glVertex2f(4.64,5.03);
glVertex2f(4.83,5.27);
glVertex2f(4.34,5.3);

glVertex2f(4.78,5.87);
glVertex2f(5.5,6);
glVertex2f(4.83,5.27);
glVertex2f(4.34,5.3);

glVertex2f(4.78,5.87);
glVertex2f(5.5,6);
glVertex2f(5.5,6.27);
glVertex2f(4.85,6.25);

glVertex2f(4.78,5.87);
glVertex2f(4.85,6.25);
glVertex2f(5.69,6.48);
glVertex2f(5.5,6.27);

glVertex2f(5.69,6.48);
glVertex2f(6.5,6.13);
glVertex2f(5.5,6);
glVertex2f(5.5,6.27);

glVertex2f(4.78,5.87);
glVertex2f(4.85,6.25);
glVertex2f(4.23,5.71);
glVertex2f(4.34,5.3);

```

```

glVertex2f(4.1,4.4);
glVertex2f(4.1,4.71);
glVertex2f(4.23,5.71);
glVertex2f(4.34,5.3);

///// Daun Tengah Atas
glVertex2f(4.1,4.71);
glVertex2f(3.97,5.6);
glVertex2f(4.17,6);
glVertex2f(4.23,5.71);

glVertex2f(3.95,6.45);
glVertex2f(3.97,5.6);
glVertex2f(4.17,6);
glVertex2f(4.03,6.61);

glVertex2f(3.95,6.45);
glVertex2f(3.97,5.6);
glVertex2f(3.82,5.7);
glVertex2f(3.8,6.5);

glVertex2f(3.71,4.88);
glVertex2f(3.97,5.6);
glVertex2f(3.82,5.7);
glVertex2f(3.76,5.41);

glVertex2f(3.71,4.88);
glVertex2f(3.97,5.6);
glVertex2f(4.1,4.71);
glVertex2f(3.92,4.78);

glVertex2f(3.95,6.45);
glVertex2f(3.8,6.5);
glVertex2f(3.65,7.03);
glVertex2f(4.03,6.61);

///// Daun Kiri Atas
glVertex2f(3.82,5.7);
glVertex2f(3.76,5.41);
glVertex2f(3.55,5.76);
glVertex2f(3.5,6.23);

glVertex2f(3.27,6.43);
glVertex2f(3.25,6.25);
glVertex2f(3.55,5.76);
glVertex2f(3.5,6.23);

glVertex2f(3.71,4.88);
glVertex2f(3.76,5.41);
glVertex2f(3.55,5.76);
glVertex2f(3.31,5.84);

glVertex2f(3.12,6.21);
glVertex2f(3.25,6.25);
glVertex2f(3.55,5.76);

```

```

        glVertex2f(3.31,5.84);

        glVertex2f(3.12,6.21);
        glVertex2f(3.25,6.25);
        glVertex2f(3.27,6.43);
        glVertex2f(2.78,6.54);

        //// Daun Kiri Tengah Atas
        glVertex2f(3.71,4.88);
        glVertex2f(2.73,5.66);
        glVertex2f(2.75,6.08);
        glVertex2f(3.31,5.84);

        glVertex2f(3.71,4.88);
        glVertex2f(2.73,5.66);
        glVertex2f(2.19,5.47);
        glVertex2f(2.84,4.94);

        glVertex2f(1.74,5.88);
        glVertex2f(2.73,5.66);
        glVertex2f(2.19,5.47);
        glVertex2f(1.37,5.59);

        glVertex2f(1.74,5.88);
        glVertex2f(2.73,5.66);
        glVertex2f(2.75,6.08);
        glVertex2f(1.64,6.18);

        glVertex2f(1.74,5.88);
        glVertex2f(1.37,5.59);
        glVertex2f(0.68,5.56);
        glVertex2f(1.64,6.18);

        glEnd();
        glPopMatrix();
    }

    // Fungsi untuk membuat Pohon jenis 2
    void Pohon2() {
        glPushMatrix();
        glRotatef(180,0,1,0);
        Pohon1(96, 40, 120);
        glPushMatrix();
        glTranslatef(-29,0,29);
        glRotatef(-90,0,1,0);
        Pohon1(96, 40, 120);
        glTranslatef(92,0,-58);
        glRotatef(-180,0,1,0);
        Pohon1(96, 40, 120);
        glPopMatrix();
        glPopMatrix();
    }
}

```

Pada program yang kami buat terdapat 2 jenis pohon, kedua pohon tersebut dipisah kedalam 2 procedure yang bernama procedure Pohon1() dan Pohon2(). Procedure Pohon1()

menggunakan GL_QUADS untuk membentuk objek pohon, dan procedure Pohon2() memanggil kembali procedure Pohon1() dan mengubahnya menjadi jenis pohon yang baru.

```
float posZPohon = -100;
int ujungP = 0;

void PohonMove()
{
    if(ujungP == 0)
    {
        posZPohon -= speed; //Kecepatan jalan
    } else
    {
        posZPohon = -100;
        ujungP = 0;
    }

    if(posZPohon <= -400) ujungP = 1;
}
```

Untuk membuat pergerakan pada pohon, terdapat procedure PohonMove() yang menggunakan conditional seperti source code di atas.

3.3 Gedung

```
void Gedung()
{
    glPushMatrix();
    glRotatef(90,0,1,0);
    glTranslatef(0,0,60);
    glScalef(6,6,6);
    glBegin(GL_QUADS);
        // Bagian Bawah
        glColor3ub(90, 185, 168);
        glVertex2f(1,0);
        glVertex2f(6,0);
        glVertex2f(6,3);
        glVertex2f(1,3);

        // Bagian 2
        glVertex2f(1,3);
        glVertex2f(6,3);
        glVertex2f(6,6);
        glVertex2f(1,6);

        // Bagian 3
        glVertex2f(1,6);
        glVertex2f(6,6);
        glVertex2f(6,9);
        glVertex2f(1,9);

        // Bagian 4
        glVertex2f(1,9);
        glVertex2f(6,9);
    glEnd();
}
```

```

glVertex2f(6,12);
glVertex2f(1,12);

// Bagian Atas
glVertex2f(1,12);
glVertex2f(6,12);
glVertex2f(6,15);
glVertex2f(1,15);

// Sela Sela Gedung
glColor3ub(30, 96, 110);
glVertex2f(-1,0);
glVertex2f(1,0);
glVertex2f(1,15);
glVertex2f(-1,15);

glVertex2f(8,0);
glVertex2f(6,0);
glVertex2f(6,15);
glVertex2f(8,15);

// Bagian Pintu Masuk
glColor3ub(196, 240, 194);
glVertex3f(2,3,-.1);
glVertex3f(2,2,-.1);
glVertex3f(5,2,-.1);
glVertex3f(5,3,-.1);
    // Pintu 1
glColor3ub(30, 96, 110);
glVertex3f(2.5,2,-.1);
glVertex3f(2.5,0,-.1);
glVertex3f(3.5,0,-.1);
glVertex3f(3.5,2,-.1);
    // Pintu 2
glColor3ub(45, 27, 0);
glVertex3f(4.5,2,-.1);
glVertex3f(4.5,0,-.1);
glVertex3f(3.5,0,-.1);
glVertex3f(3.5,2,-.1);

// Jendela Gedung
glColor3ub(196, 240, 194);
glVertex3f(2,6,-.1);
glVertex3f(2,4,-.1);
glVertex3f(3,4,-.1);
glVertex3f(3,6,-.1);

glVertex3f(4,6,-.1);
glVertex3f(4,4,-.1);
glVertex3f(5,4,-.1);
glVertex3f(5,6,-.1);

glVertex3f(2,9,-.1);
glVertex3f(2,7,-.1);
glVertex3f(3,7,-.1);
glVertex3f(3,9,-.1);

```

```

        glVertex3f(4,9,-.1);
        glVertex3f(4,7,-.1);
        glVertex3f(5,7,-.1);
        glVertex3f(5,9,-.1);

        glVertex3f(2,12,-.1);
        glVertex3f(2,15,-.1);
        glVertex3f(3,15,-.1);
        glVertex3f(3,12,-.1);

        glVertex3f(4,12,-.1);
        glVertex3f(4,15,-.1);
        glVertex3f(5,15,-.1);
        glVertex3f(5,12,-.1);
    glEnd();
    glPopMatrix();
}

```

Lalu terdapat procedure Gedung() untuk membuat bagian gedung, untuk membuat gedung tersebut menggunakan GL_QUADS. Bagian-bagian tersebut membuat pintu, jendela, dan sela-sela dari semua gedung.

```

void GedungMove() {
    if(ujungG == 0)
    {
        posZGedung -= speed; //Kecepatan jalan
    } else
    {
        posZGedung = -100;
        ujungG = 0;
    }

    if(posZGedung <= -400) ujungG = 1;
}

```

Untuk membuat pergerakan pada gedung, terdapat procedure PohonMove() yang menggunakan conditional seperti source code di atas.

3.4 Lantai

```

void Environment()
{
    //Grid bawah
    glPushMatrix();
    glEnable(GL_LINE_SMOOTH);
    glColor3ub(100,100,255);
    glLineWidth(1);
    glBegin(GL_LINES);
    for(i=Z_MIN;i<Z_MAX;i+=gap)
    {
        glVertex3f(i, 0, Z_MIN);
        glVertex3f(i, 0, Z_MAX);
    }
}

```

```

    }
    for(i=X_MIN;i<X_MAX;i+=gap)
    {
        glVertex3f(X_MIN, 0, i);
        glVertex3f(X_MAX, 0, i);
    }
    glEnd();
glPopMatrix();

glPushMatrix();
glTranslatef(0,0,500);
glScalef(0.5,0.69,0.7);
    //box depan
    glPushMatrix();
    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, _textureIdDepan);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER,
GL_LINEAR);
    glColor3f(1,1,1);
    glBegin(GL_QUADS);
    glTexCoord2f(0.0f, 0.0f);
    glVertex3f(100.0f, 0.05f, 100.0f);
    glTexCoord2f(1.0f, 0.0f);
    glVertex3f(100.0f, 135, 100.0f);
    glTexCoord2f(1.0f, 1.0f);
    glVertex3f(-100.0f, 135, 100.0f);
    glTexCoord2f(0.0f, 1.0f);
    glVertex3f(-100.0f, 0.05f, 100.0f);
    glEnd();
    glPopMatrix();

    //box kiri
    glPushMatrix();
    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, _textureIdKiri);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER,
GL_LINEAR);
    glBegin(GL_QUADS);
    glTexCoord2f(0.0f, 0.0f);
    glVertex3f(200.0f, 0.05f, 100.0f);
    glTexCoord2f(1.0f, 0.0f);
    glVertex3f(200.0f, 135, 100.0f);
    glTexCoord2f(1.0f, 1.0f);
    glVertex3f(100.0f, 135, 100.0f);
    glTexCoord2f(0.0f, 1.0f);
    glVertex3f(100.0f, 0.05f, 100.0f);
    glEnd();
    glPopMatrix();

    //box kanan
    glPushMatrix();
    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, _textureIdKanan);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER,
GL_LINEAR);
    glBegin(GL_QUADS);

```



```

        glTexCoord2f(0.0f, 0.0f);
        glVertex3f(-100.0f, 0.05f, 100.0f);
        glTexCoord2f(1.0f, 0.0f);
        glVertex3f(-100.0f, 135, 100.0f);
        glTexCoord2f(1.0f, 1.0f);
        glVertex3f(-200.0f, 135, 100.0f);
        glTexCoord2f(0.0f, 1.0f);
        glVertex3f(-200.0f, 0.05f, 100.0f);
        glEnd();
        glPopMatrix();
    glPopMatrix();

    //box belakang
    glPushMatrix();
    glEnable(GL_TEXTURE_2D);
    glBindTexture(GL_TEXTURE_2D, _textureIdBelakang);
    glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER,
GL_LINEAR);
    glBegin(GL_QUADS);
    glTexCoord2f(0.0f, 0.0f);
    glVertex3f(100.0f, 0.05f, -100.0f);
    glTexCoord2f(1.0f, 0.0f);
    glVertex3f(100.0f, 135, -100.0f);
    glTexCoord2f(1.0f, 1.0f);
    glVertex3f(-100.0f, 135, -100.0f);
    glTexCoord2f(0.0f, 1.0f);
    glVertex3f(-100.0f, 0.05f, -100.0f);
    glEnd();
    glPopMatrix();

    //////////// LANTAI DI BAWAH GRID ////////////
    glPushMatrix();
        glColor3ub(10,10,10);
        glBegin(GL_QUADS);
        glVertex3f(-500.,-0.,-700.);
        glVertex3f(500.,-0.,-700.);
        glVertex3f(500.,-0.,700.);
        glVertex3f(-500.,-0.,700.);
        glEnd();
    glPopMatrix();
}

```

Pada program kami terdapat procedure Environment() yang akan menggunakan bentuk grid sebagai lantai dibawahnya, pada procedure ini juga membuat box depan, belakang, kiri, dan kanan.

3.5 Jalan dan Tembok

```

void Jalan() {
    glPushMatrix(); //Bagian kepala
        panjang = 50;
        lebar = 50;
        tinggi = 0.5;

```

```

glTranslatef(0,0,5);
glColor3f(0.1f,0.1f,0.1f);
//depan
glPushMatrix();
glTranslatef(0,0,panjang/2);
glBegin(GL_QUADS);
glVertex3f(-lebar/2,0.0f,0.0f);
glVertex3f(lebar/2,0.0f,0.0f);
glVertex3f(lebar/2,tinggi,0.0f);
glVertex3f(-lebar/2,tinggi,0.0f);
glEnd();
glPopMatrix();
// atas
glPushMatrix();
glRotated(-90, 1, 0, 0);
glTranslatef(0,0,tinggi);
glBegin(GL_QUADS);
glVertex3f(-lebar/2,-panjang/2,0.0f);
glVertex3f(lebar/2,-panjang/2,0.0f);
glVertex3f(lebar/2,panjang/2,0.0f);
glVertex3f(-lebar/2,panjang/2,0.0f);
glEnd();
glPopMatrix();
// belakang
glPushMatrix();
glRotated(-180, 1, 0, 0);
glTranslatef(0,0,panjang/2);
glBegin(GL_QUADS);
glTexCoord2f(0.0f,0.0f); // kiri bawah
glVertex3f(-lebar/2,-tinggi,0.0f);
glTexCoord2f(1.0f,0.0f); // kanan bawah
glVertex3f(lebar/2,-tinggi,0.0f);
glTexCoord2f(1.0f,1.0f); // kanan atas
glVertex3f(lebar/2,0.0f,0.0f);
glTexCoord2f(0.0f,1.0f); // kanan bawah
glVertex3f(-lebar/2,0.0f,0.0f);
glEnd();
glPopMatrix();
// bawah
glPushMatrix();
glRotated(90, 1, 0, 0);
glTranslatef(0,0,0);
glBegin(GL_QUADS);
glTexCoord2f(0.0f,0.0f); // kiri bawah
glVertex3f(-lebar/2,-panjang/2,0.0f);
glTexCoord2f(1.0f,0.0f); // kanan bawah
glVertex3f(lebar/2,-panjang/2,0.0f);
glTexCoord2f(1.0f,1.0f); // kanan atas
glVertex3f(lebar/2,panjang/2,0.0f);
glTexCoord2f(0.0f,1.0f); // kanan bawah
glVertex3f(-lebar/2,panjang/2,0.0f);
glEnd();
glPopMatrix();
// kiri
glPushMatrix();
glRotated(-90, 0, 1, 0);

```

```

        glTranslatef(0,0,lebar/2);
        glBegin(GL_QUADS);
        glVertex3f(-panjang/2,0.0f,0.0f);
        glVertex3f(panjang/2,0.0f,0.0f);
        glVertex3f(panjang/2,tinggi,0.0f);
        glVertex3f(-panjang/2,tinggi,0.0f);
        glEnd();
        glPopMatrix();
        // kanan
        glPushMatrix();
        glRotated(90, 0, 1, 0);
        glTranslatef(0,0,lebar/2);
        glBegin(GL_QUADS);
        glVertex3f(-panjang/2,0.0f,0.0f);
        glVertex3f(panjang/2,0.0f,0.0f);
        glVertex3f(panjang/2,tinggi,0.0f);
        glVertex3f(-panjang/2,tinggi,0.0f);
        glEnd();
        glPopMatrix();
    glPopMatrix();

    glPushMatrix();
        panjang = 6;
        lebar = 1;
        tinggi = 4;
        glRotated(-90,1,0,0);
        glTranslatef(0,-7.5,-2.29);
        glColor3f(1,1,1);
        //depan
        glPushMatrix();
        glTranslatef(0,0,panjang/2);
        glBegin(GL_QUADS);
        glVertex3f(-lebar/2,0.0f,0.0f);
        glVertex3f(lebar/2,0.0f,0.0f);
        glVertex3f(lebar/2,tinggi,0.0f);
        glVertex3f(-lebar/2,tinggi,0.0f);
        glEnd();
        glPopMatrix();
    glPopMatrix();
}

// Fungsi untuk membuat TEMBOK
void Tembok() {
    glPushMatrix();
    glColor3ub(100,100,100);
        glPushMatrix();
            glTranslatef(5, 0, 70);
            glScalef(3, 8, 1);
            glutSolidCube(3);
            glPopMatrix();
        glEnable(GL_TEXTURE_2D);
        glBindTexture(GL_TEXTURE_2D, _textureId);
        glTexParameterf(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER,
GL_LINEAR);
        glScalef(0.75,1.5,1);
        glTranslatef(7.7,0,68.4);

```

```

glBegin(GL_QUADS);
glTexCoord2f(0,0);
glVertex2f(-8,0);
glTexCoord2f(1,0);
glVertex2f(5,0);
glTexCoord2f(1,1);
glVertex2f(5,8);
glTexCoord2f(0,1);
glVertex2f(-8,8);
glEnd();
glPopMatrix();
}

```

Terdapat procedure Jalan() untuk membuat jalanan dan juga garis marka jalan. Kemudian ditengah jalan nanti akan terdapat 2 objek dari procedure Tembok(), objek ini berupa gambar ‘Among Us’ yang dapat ditabrak.

```

void PergerakanJalan()
{
    //jika jalan belum mencapai ujung
    if(ujung == 0)
    {
        posZJalan -= speed; //Kecepatan jalan
    } else
    {
        posZJalan = -100;
        ujung = 0;
    }

    if(posZJalan <= -400) {
        ujung = 1;
    }
    if(speed >= 10){
        speed = 9;
    }
}

void TembokMove(){
    if(ujungT == 0)
    {
        posZTembok -= speed; //Kecepatan jalan
    } else
    {
        posZTembok = 100;
        ujungT = 0;
    }

    if(posZTembok <= -400) ujungT = 1;
}

```

Untuk mengatur pergerakan jalan dan juga pergerakan objek Tembok(), terdapat procedure PergerakanJalan() dan TembokMove() yang menggunakan banyak conditional agar dapat bergerak.

3.6 IsTouch dan Game Over

```
bool touch = false;

void isTouch(){
    if ((posZTruk < posZTembok+80 and posZTruk > posZTembok +64)
and (posXTruk >= -8 and posXTruk <= 5)){
        touch = true;
    }
}

void drawStrokeText2(char* str,int x,int y,int z)
{
    char *c;
    glPushMatrix();
    glTranslatef(x, y+8,z);
    glScalef(0.05f,0.05f,0.02f);

    for (c=str; *c != '\0'; c++) {
        glutStrokeCharacter(GLUT_STROKE_MONO_ROMAN , *c);
    }
    glPopMatrix();
}

void display() {
    // Kalau move dan angle tidak nol, gerakkan kamera...
    if (deltaMove)
        moveMeFlat(deltaMove);
        if (deltaAngle) {
            angle += deltaAngle;
            orientMe(angle);
        }
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glPushMatrix();
    glRotated(rotAngleX+10, 1, 0, 0);
    glRotated(rotAngleY, 0, 1, 0);
    glRotated(rotAngleZ, 0, 0, 1);
    isTouch();
    if (touch){
        TrukDepan();
        TrukBelakang();
        drawStrokeText2("Game Over",-23,3,0);
        glRotated(180,0,1,0);
        drawStrokeText2("Game Over",-23,10,0);
        rotAngleY-=1;
        glClearColor(1,1,1,1);
    } else {
        Environment();
        Object();
        glClearColor(0,0,0,1);
    }
    glPopMatrix();
    glFlush();
    glutSwapBuffers();
}
```

Di atas tadi sudah dibuat procedure Tembok() yang akan memunculkan objek gambar 'Among Us' yang dapat ditabrak, pada procedure isTouch akan membuat boolean apakah truk menabrak objek tembok.

Terdapat procedure drawStrokeText2() yang memiliki parameter str, x, y, z, procedure ini akan membuat tampilan Game Over apabila truk yang kita kendalikan menabrak objek tembok berupa 'Among Us'.

Lalu terdapat procedure display() yang akan mendisplay arah kamera dan membuat conditional apabila truk menabrak objek 'Among Us' yang akan memunculkan tampilan Game Over.

3.7 Object

```
void Object()
{
    glPushMatrix();
        glPushMatrix();
            TrukMove();
            glRotatef(angTrukDepan,0,1,0);
            glTranslatef(posXTruk,0, posZTruk);
            TrukDepan();
        glPopMatrix();
        glPushMatrix();
            TrukMove2();
            glRotatef(angTrukBelakang,0,1,0);
            glTranslatef(posXTruk,0, posZTruk);
            TrukBelakang();
        glPopMatrix();
    glPopMatrix();

    glPushMatrix();
    int a;
    PergerakanJalan();
    for(a = 180; a > 0; a -= 1){
        glPushMatrix();
            glTranslatef(-4,0,posZJalan+(a*5));
            Jalan();
        glPopMatrix();
    }
    glPopMatrix();

    glPushMatrix();
    int b;
    PohonMove();
    for(b = 20; b > 0; b -= 1){
        glPushMatrix();
            glTranslatef(-4,0,posZPohon+(b*50));
            Pohon1(96, 40, 120);
            Pohon2();
            Gedung();
            glPushMatrix();
            glTranslatef(0,0,-100);
```

```

        glRotated(180,0,1,0);
        Gedung();
        glPopMatrix();
    glPopMatrix();
}
glPopMatrix();

glPushMatrix();
    TembokMove();
    glTranslatef(-4,0,posZTembok);
    Tembok();
    glPushMatrix();
    glTranslatef(-15,0,0);
    Tembok();
    glPopMatrix();
glPopMatrix();
}

```

Terdapat procedure Object() yang akan menampilkan semua objek ketika kita melakukan run pada program ini.

BAB IV

KESIMPULAN

4.1 Kesimpulan

Berdasarkan hasil dari praktikum dan tugas yang kami buat, dapat disimpulkan bahwa pembuatan Truk animasi ini dapat dilakukan menggunakan OpenGL. Selain itu, GLUT yang merupakan library tambahan pada OpenGL juga digunakan dalam membuat animasi Truk ini. Fungsi Fungsi yang digunakan dalam membuat animasi adalah fungsi pembentukan objek, lighting, pengaturan camera dan lain sebagainya. Aspek yang digunakan pada OpenGL ini adalah depth dan lighting, texture mapping, dan juga interaksi objek pada OpenGL.

DAFTAR PUSTAKA

<http://www.dufatancom.id/2019/08/pengertian-opengl-dan-cara-kerjanya.html> diakses pada tanggal 26 Mei 2022 pukul 22:20 WIB

<http://muhammadfarizal01.blogspot.com/2015/05/pengenalan-opengl-glut.html> diakses pada tanggal 26 Mei 2022 pukul 22:23 WIB

<https://media.neliti.com/media/publications/174291-ID-pembentukan-transformasi-objek-2-dimensi.pdf> diakses pada tanggal 26 Mei 2022 pukul 22:28 WIB

<http://senigaming.blogspot.com/2017/05/pengertian-opengl-dan-cara-kerjanya.html> diakses pada tanggal 26 Mei 2022 pukul 22:30 WIB

LAMPIRAN

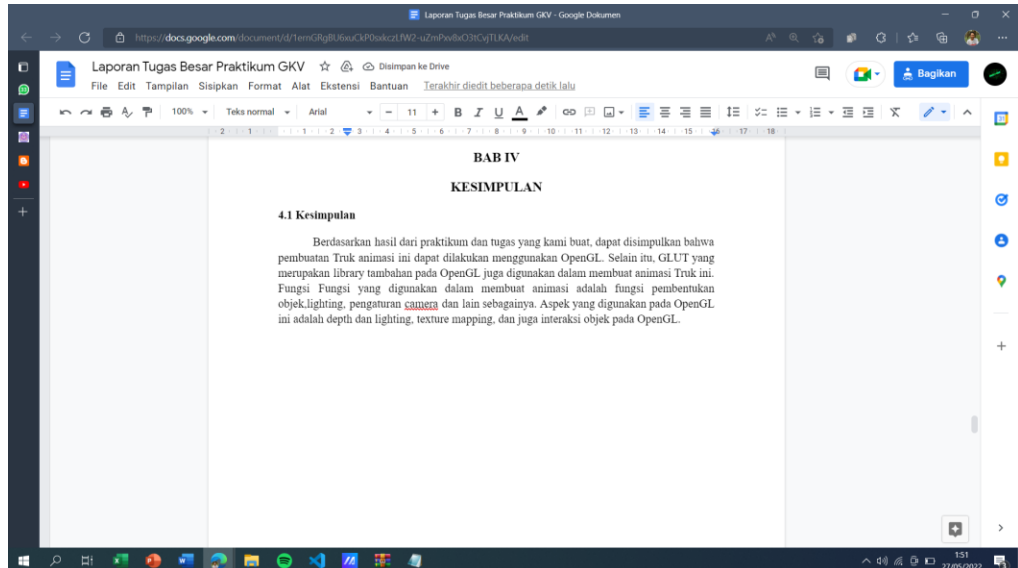
A. Link Video : <https://www.youtube.com/watch?v=R-BHNa8RBkk&t=363s>

B. Struktur Pembagian Tugas

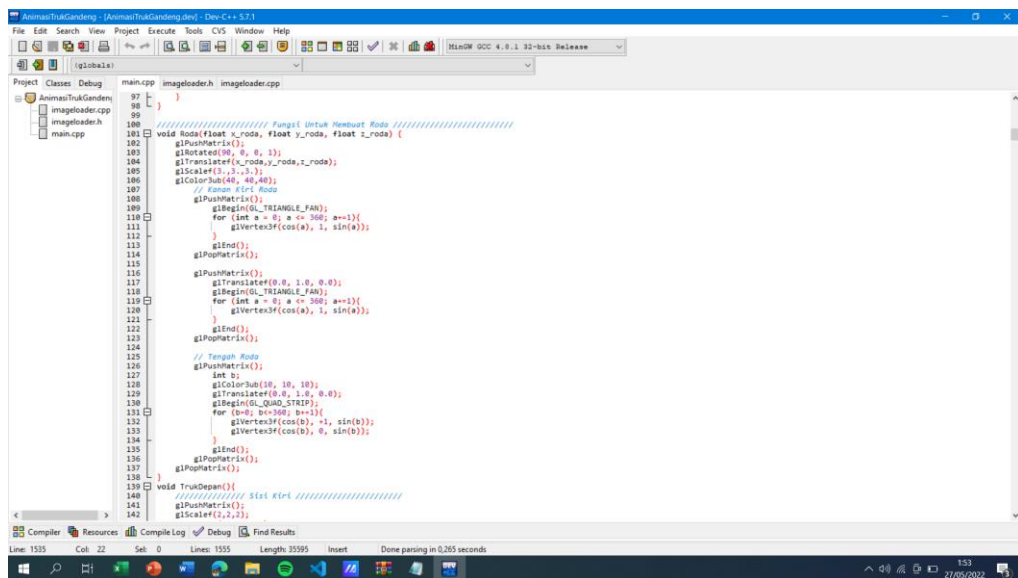
1. Rizal Zeri Subakti - 24060120130062

Uraian Tugas :

- Membuat laporan BAB 4



- Merancang fungsi Roda,Truk Depan, Truk Belakang,Trukmove,Trukmove2 pada Animasi Truk Roda



Truk depan

```

AnimasTrukGandeng - AnimasTrukGandeng.sln - Dev-C++ 5.7.1
File Edit Search View Project Execute Tools CVS Window Help
[Icons] [global]
Project: Classes: Debug
main.cpp imageloder.h imageloder.cpp
136
137
138
139 ///////////////////////////////////////////////////
140 void TrukDepan()
141 {
142     glPushMatrix();
143     glScalef(2,2,2);
144     glRotated(-90,0,1,0);
145     glTranslatef(-10,-0.7,2);
146     glColor3ub(20,20,20);
147     glBegin(GL_TRIANGLE_STRIP);
148     // Face
149     glVertex2f(4,4,3);
150     glVertex2f(4,3,3);
151     glVertex2f(6,4,3);
152     glVertex2f(6,3,3);
153     glVertex2f(8,4,3);
154     glVertex2f(8,3,3);
155     glEnd();
156     // Body
157     glColor3ub(100,100,100);
158     glBegin(GL_QUADS);
159     // Bawah Belakang
160     glVertex2f(0,4);
161     glVertex2f(8,5,2,4);
162     glVertex2f(4,2,3);
163     glVertex2f(4,3,3);
164     // Bawah Tengah
165     glVertex2f(4,2,3);
166     glVertex2f(4,3,3);
167     glVertex2f(12,2,3);
168     // Bawah Depan
169     glVertex2f(12,3,3);
170     glVertex2f(12,2,2);
171     glVertex2f(14,2,2);
172     glVertex2f(14,3);
173     // Depan Atas
174     glVertex2f(1,3,3);
175     glVertex2f(14,3);
176     glVertex2f(8,3);
177     glVertex2f(8,4,7);
178     // Atas Tengah
179     glVertex2f(8,3);
180     glVertex2f(8,4,7);
181     glVertex2f(6,4,3);
182

```

Truk Belakang

```

AnimasTrukGandeng - AnimasTrukGandeng.sln - Dev-C++ 5.7.1
File Edit Search View Project Execute Tools CVS Window Help
[Icons] [global]
Project: Classes: Debug
main.cpp imageloder.h imageloder.cpp
364
365
366 void TrukBelakang()
367 {
368     // Kiri
369     glPushMatrix();
370     glTranslatef(-5,0,-23);
371     glScalef(2,2,2);
372     glRotated(-90,0,1,0);
373     glColor3ub(55,55,55);
374     glBegin(GL_TRIANGLE_STRIP);
375     glVertex3f(-1,3,0); // Titik Tengah Depan
376     glVertex3f(-4,5,7,0); // Titik Atas Depan
377     glVertex3f(-2,5,2,0); // Titik Bawah Depan
378     glVertex3f(-14,7,0); // Titik Atas Belakang
379     glVertex3f(-12,2,0); // Titik Bawah Tengah
380     glVertex3f(-14,2,0); // Titik Bawah Belakang
381     glEnd();
382     // Outline Kiri
383     glBegin(GL_QUADS);
384     glColor3ub(155,155,155);
385     // Bawah
386     glVertex3f(-15,1,0);
387     glVertex3f(-15,2);
388     glVertex3f(-1,2,0);
389     glVertex3f(-2,1,0);
390     // Depan Bawah
391     glVertex3f(-2,5,2,0);
392     glVertex3f(-1,2,0);
393     glVertex3f(8,1,0);
394     glVertex3f(-1,5,4);
395     // Depan Atas
396     glVertex3f(8,0,0);
397     glVertex3f(-1,5,0);
398     glVertex3f(-4,5,7,0);
399     glVertex3f(-4,6,0);
400     // Atas
401     glVertex3f(-4,5,7,0);
402     glVertex3f(-4,8,0);
403     glVertex3f(-15,8,0);
404     glVertex3f(-14,7,0);
405     // Belakang
406     glVertex3f(-15,8,0);
407     glVertex3f(-14,7,0);
408     glVertex3f(-14,2,0);
409     glVertex3f(-15,2,0);
410

```

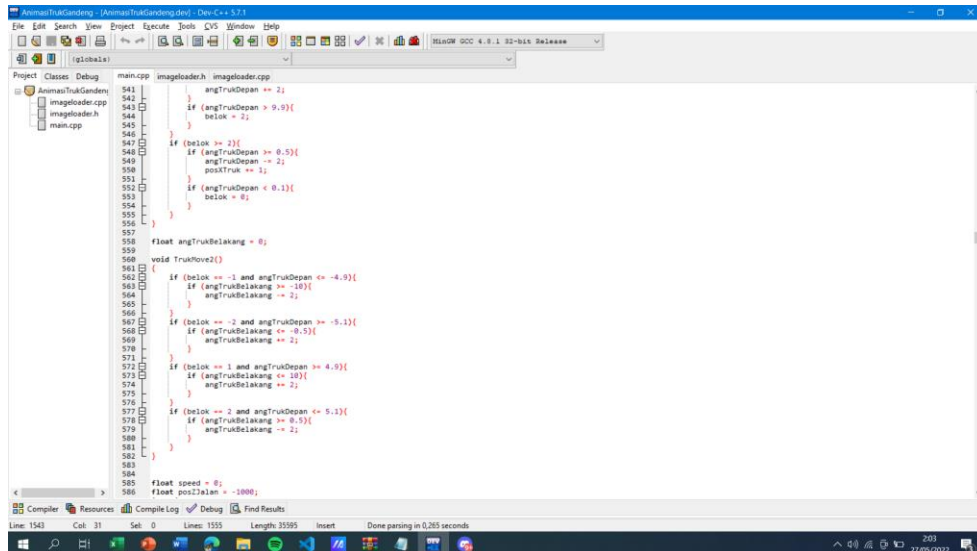
Trukmove

```

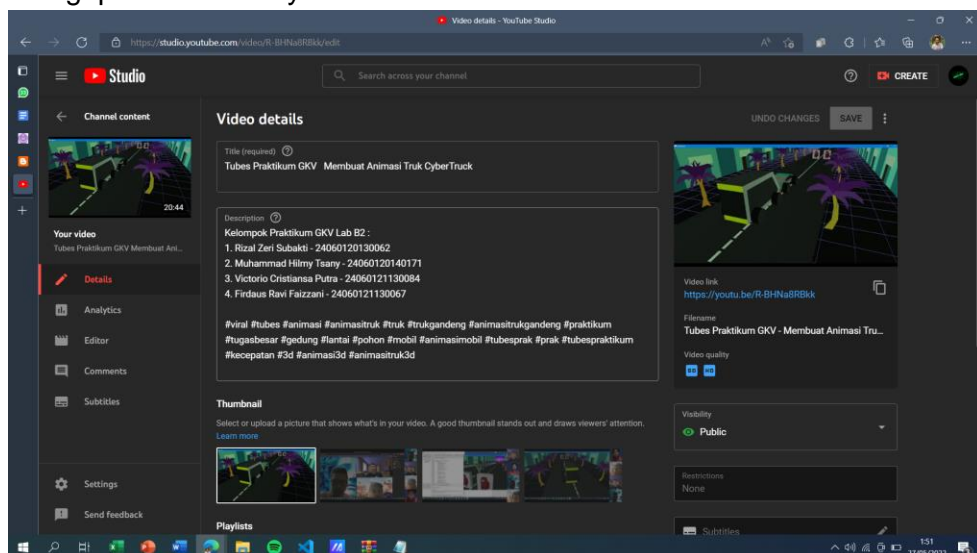
AnimasTrukGandeng - AnimasTrukGandeng.sln - Dev-C++ 5.7.1
File Edit Search View Project Execute Tools CVS Window Help
[Icons] [global]
Project: Classes: Debug
main.cpp imageloder.h imageloder.cpp
514
515 float posXTruk = 0.0, posYTruk = 0.0;
516
517 void TrukMove()
518 {
519     //Belok kanan
520     if (belok == -1){
521         if (angTrukDepan <= -10){
522             angTrukDepan += 2;
523             if (angTrukDepan < -9.9){
524                 belok = -1;
525             }
526         }
527     }
528     //Belok kiri
529     if (belok == 1){
530         if (angTrukDepan <= 0.5){
531             angTrukDepan += 2;
532             posXTruk += 1;
533             if (angTrukDepan > 0.1){
534                 belok = 0;
535             }
536         }
537     }
538     //Belok kiri
539     if (belok == 1){
540         if (angTrukDepan <= 10){
541             angTrukDepan += 2;
542             if (angTrukDepan > 9.9){
543                 belok = 2;
544             }
545         }
546     }
547     //Belok kanan
548     if (belok == 2){
549         if (angTrukDepan <= 0.5){
550             angTrukDepan += 2;
551             posXTruk += 1;
552             if (angTrukDepan < 0.1){
553                 belok = 0;
554             }
555         }
556     }
557     float angTrukBelakang = 0;
558

```

Trukmove2



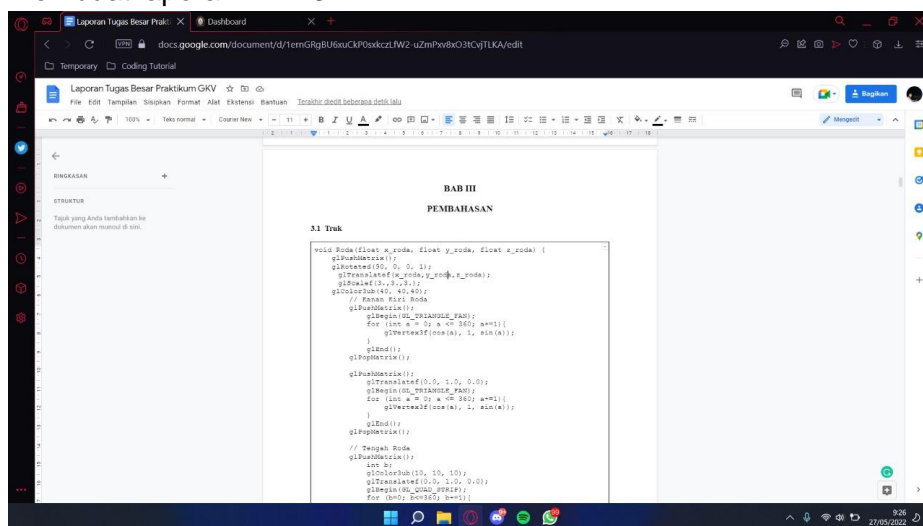
- Mengupload video di youtube



2. Muhammad Hilmy Tsany - 24060120140171

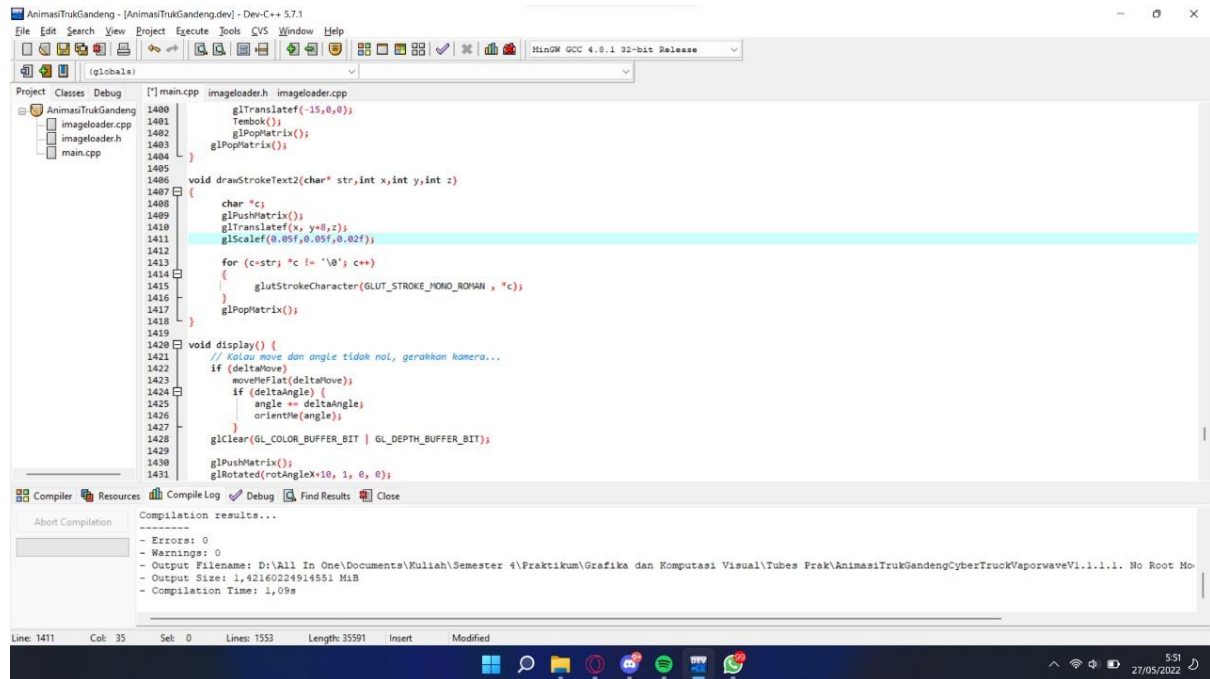
Uraian Tugas :

- Membuat laporan BAB 3



- Merancang fungsi environment, istouch, object, drawstrokertext2 Environment

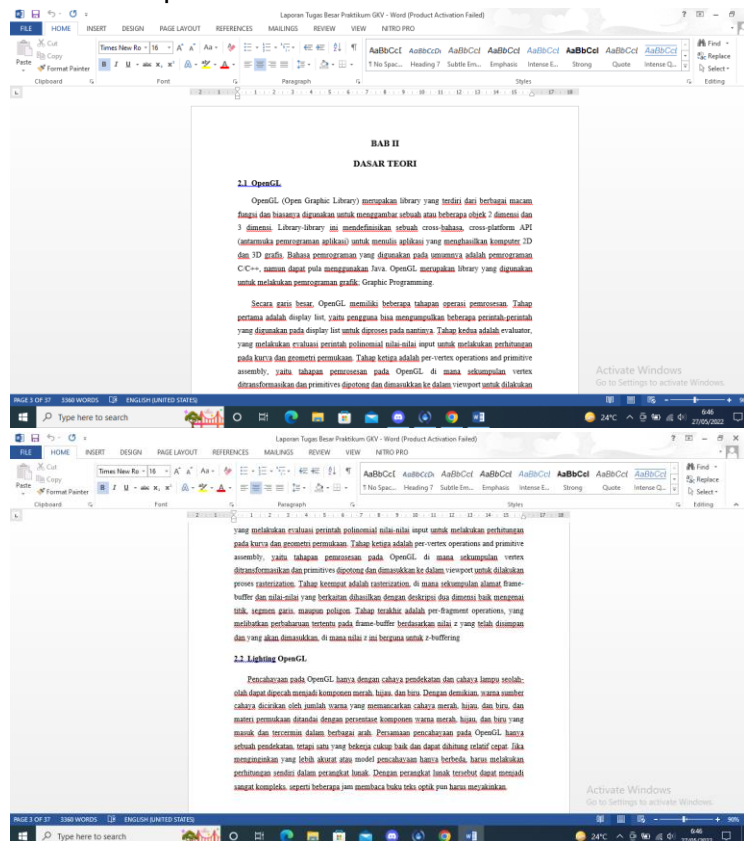


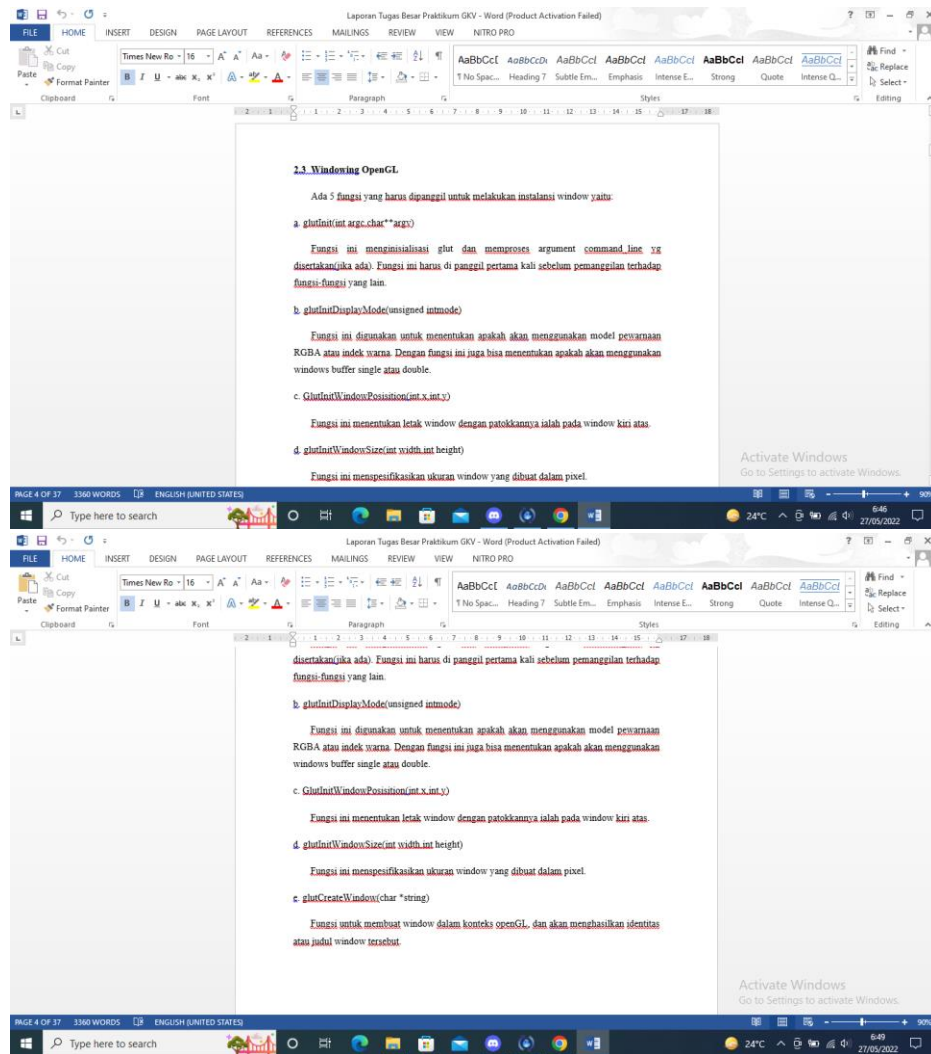


3. Victorio Cristiansa Putra - 24060121130084

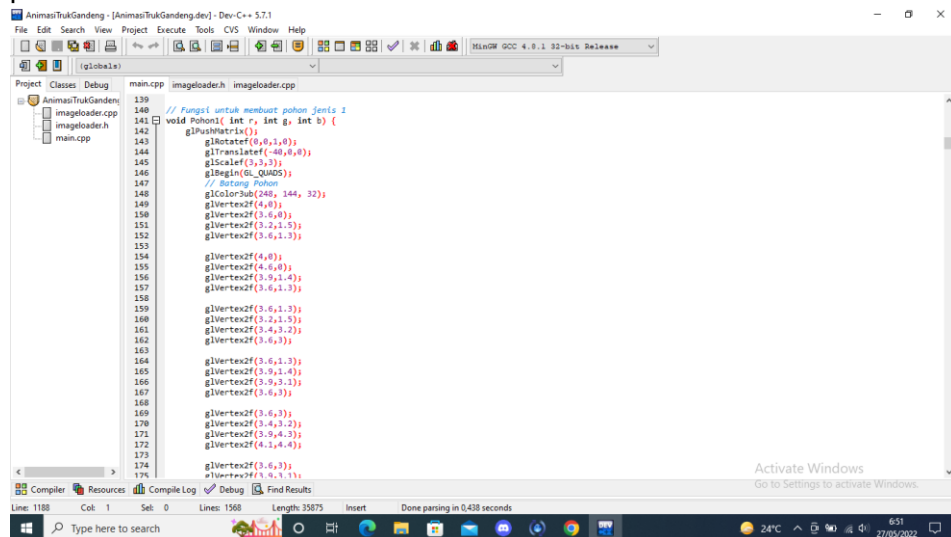
Uraian Tugas :

- Membuat laporan BAB 2





- Membuat fungsi pohon1, pohon2, gedung, gedungmove, pohonmove, pohon1



pohon2

```

main.cpp | imageloder.h | imageloder.cpp
426 | glVertex2f(2.75,6.88);
427 | glVertex2f(1.64,6.18);
428 |
429 | glVertex2f(1.74,5.88);
430 | glVertex2f(1.37,5.59);
431 | glVertex2f(0.68,5.56);
432 | glVertex2f(1.64,6.18);
433 |
434 | glEnd();
435 | glPopMatrix();
436 |
437 |
438 | // Fungsi untuk membuat Pohon jenis 2
439 | void Pohon2() {
440 |     glPushMatrix();
441 |     glRotatef(180,0,1,0);
442 |     Pohon1(90, 40, 120);
443 |     glPopMatrix();
444 |     glTranslatef(-29,0,29);
445 |     glRotatef(-90,0,1,0);
446 |     Pohon1(90, 40, 120);
447 |     glTranslatef(92,0,-50);
448 |     glRotatef(-180,0,1,0);
449 |     Pohon1(90, 40, 120);
450 |     glPopMatrix();
451 | }
452 |
453 |
454 | float yappari = 15;
455 |
456 |
457 | void TrukDepan() {
458 |     // Sisi Kiri
459 |     glPushMatrix();
460 |     glScalef(2,2,2);
461 |     glRotatef(-90,0,1,0);
462 |     glTranslatef(-10,-0.7,1);

```

gedung

```

main.cpp | imageloder.h | imageloder.cpp
951 | }
952 |
953 | void Gedung()
954 | {
955 |     glPushMatrix();
956 |     glRotatef(90,0,1,0);
957 |     glTranslatef(0,0,60);
958 |     glScalef(6,6,6);
959 |     glBegin(GL_QUADS);
960 |     // Bagian South
961 |     glColor3ub(90, 185, 168);
962 |     glVertex2f(1,0);
963 |     glVertex2f(6,0);
964 |     glVertex2f(6,3);
965 |     glVertex2f(1,3);
966 |
967 |     // Bagian 2
968 |     glVertex2f(1,3);
969 |     glVertex2f(6,3);
970 |     glVertex2f(6,6);
971 |     glVertex2f(1,6);
972 |
973 |     // Bagian 3
974 |     glVertex2f(1,6);
975 |     glVertex2f(6,6);
976 |     glVertex2f(6,9);
977 |     glVertex2f(1,9);
978 |
979 |     // Bagian 4
980 |     glVertex2f(1,9);
981 |     glVertex2f(6,9);
982 |     glVertex2f(6,12);
983 |     glVertex2f(1,12);
984 |
985 |     // Bagian Atas
986 |     glVertex2f(1,12);
987 |     glVertex2f(6,12);

```

gedungmove + pohon move

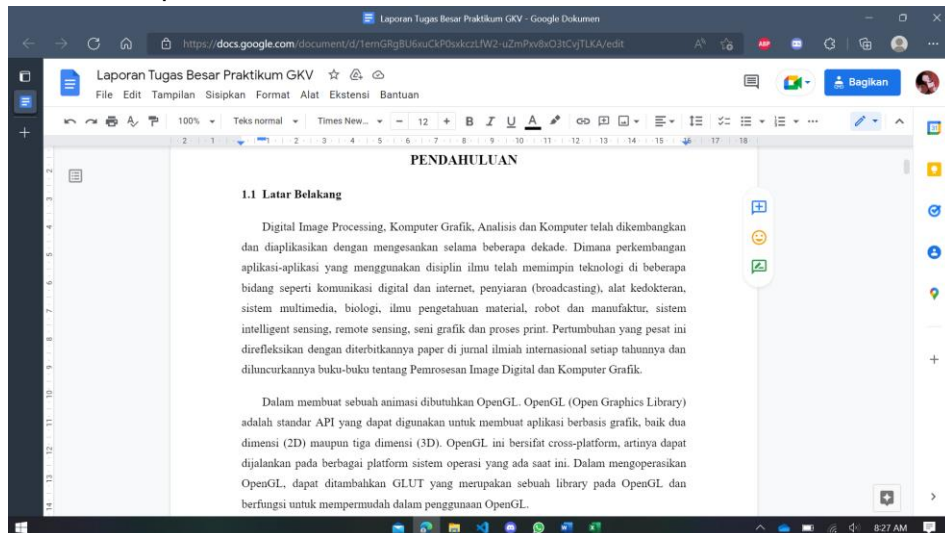
```

main.cpp | imageloder.h | imageloder.cpp
1291 | float pos2Pohon = -100;
1292 | int ujungP = 0;
1293 |
1294 | void PohonMove()
1295 | {
1296 |     if(ujungP == 0)
1297 |     {
1298 |         pos2Pohon -= speed; //Kecepatan jalan
1299 |     }
1300 |     else
1301 |     {
1302 |         pos2Pohon = -100;
1303 |         ujungP = 0;
1304 |     }
1305 |     if(pos2Pohon <= -400) ujungP = 1;
1306 | }
1307 |
1308 | float pos2Gedung = -100;
1309 | int ujungG = 0;
1310 |
1311 | void GedungMove()
1312 | {
1313 |     if(ujungG == 0)
1314 |     {
1315 |         pos2Gedung -= speed; //Kecepatan jalan
1316 |     }
1317 |     else
1318 |     {
1319 |         pos2Gedung = -100;
1320 |         ujungG = 0;
1321 |     }
1322 |     if(pos2Gedung <= -400) ujungG = 1;
1323 | }
1324 | float pos2Tembok = 100;
1325 | int ujungT = 0;
1326 |
1327 | void TembokMove()

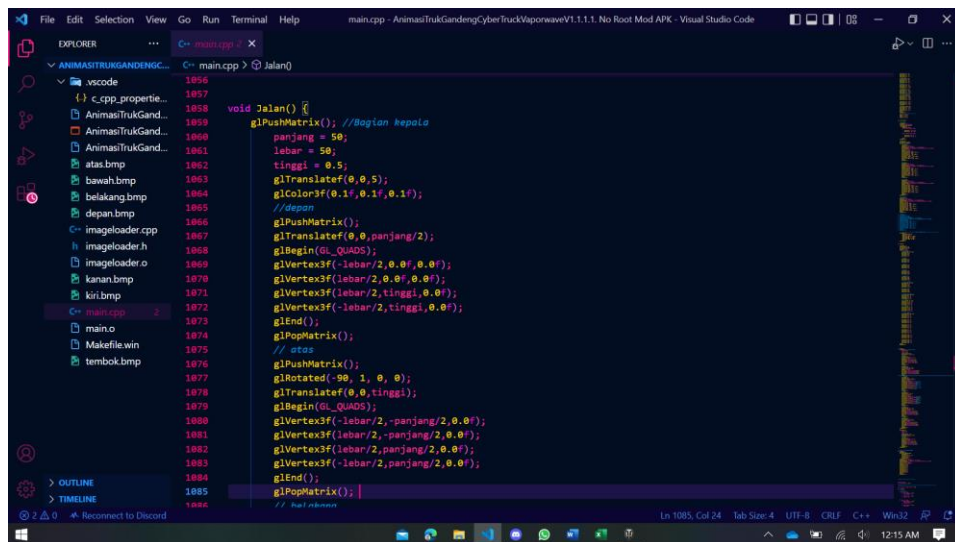
```

4. Firdaus Ravi Faizzani - 24060121130067
Uraian Tugas :

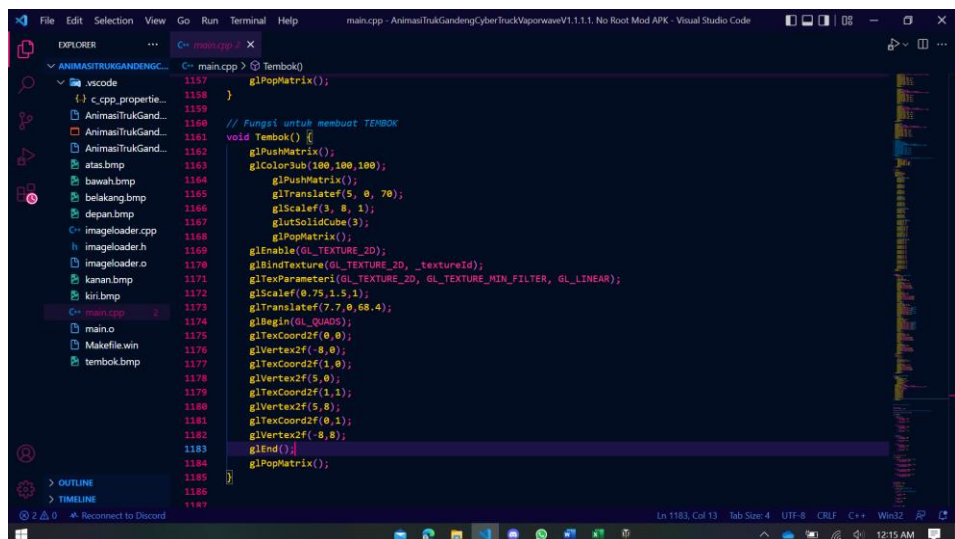
- Membuat laporan BAB 1



- Membuat fungsi jalan, tembok, pergerakanjalan, tembokmove
Jalan



Tembok



PergerakanJalan

```
1264 }
1265
1266 float speed = 0;
1267 float posZJalan = -1000;
1268 int ujung = 0;
1269
1270 void PergerakanJalan()
1271 {
1272     //jika jalan belum mencapai ujung
1273     if(ujung == 0)
1274     {
1275         posZJalan -= speed; //Kecepatan jalan
1276     } else
1277     {
1278         posZJalan = -100;
1279         ujung = 0;
1280     }
1281
1282     if(posZJalan <= -400) {
1283         ujung = 1;
1284     }
1285     if(speed >= 10){
1286         speed = 0;
1287     }
1288 }
1289
1290 float posZPohon = -100;
1291 int ujungP = 0;
1292
1293 void PohonMove()
```

TembokMove

```
1317 posZGedung = -100;
1318 ujungG = 0;
1319
1320 if(posZGedung <= -400) ujungG = 1;
1321
1322 float posZTembok = 100;
1323 int ujungT = 0;
1324
1325 void TembokMove()
1326 {
1327     if(ujungT == 0)
1328     {
1329         posZTembok -= speed; //Kecepatan jalan
1330     } else
1331     {
1332         posZTembok = 100;
1333         ujungT = 0;
1334     }
1335
1336     if(posZTembok <= -400) ujungT = 1;
1337 }
1338
1339 bool touch = false;
1340
1341 void isTouch()
1342 {
1343     if ((posZTruk < posZTembok+80 and posZTruk > posZTembok +64) and (posXTruk >= -8 and posXTruk <= 5)){
1344         touch = true;
1345     }
1346 }
1347
```