

Tugas Besar II IF2123 Aljabar Geometri
Simulasi Transformasi Linier pada Bidang 2D dan 3D
dengan Menggunakan OpenGL API



Oleh:

Irfan Haris Widyadhana (13517041)

Naufal Zhafran Latif (13517095)

Fajar Muslim (13517149)

INSTITUT TEKNOLOGI BANDUNG
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
PROGRAM STUDI TEKNIK INFORMATIKA
2018

Kata Pengantar

Pertama kami memanjatkan puji syukur ke hadirat Allah SWT karena rahmat, karunia serta hidayat-Nya, sehingga kami bisa menyelesaikan laporan Tugas Besar II Aljabar Geometri dapat kami susun hingga selesai.

Laporan ini dibuat untuk penyelesaian tugas besar mata kuliah Aljabar Geometri Semester I tahun akademik 2018/2019 di Program Studi Informatika Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung. Laporan ini dapat terselesaikan berkat bantuan dari berbagai pihak, sehingga penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan bimbingan, dorongan doa sehingga tugas dan laporan ini berhasil kami selesaikan.

Dalam pengerjaan makalah ini penulis mendapat berbagai hambatan diantaranya tugas ini dikerjakan bersamaan dengan tugas besar mata kuliah lainnya yang batas pengumpulannya hampir berdekatan sehingga harus pandai membagi waktu.

Penulis menyadari bahwa laporan ini masih belum sempurna terutama karena urusan akademik yang lain, sehingga masih banyak kekurangan pada tata bahasanya dan isinya. Oleh karena itu, penulis mengharapkan adanya kritik dan saran yang membangun untuk perubahan yang lebih baik.

Bandung, 18 November 2018

Tim Penulis

BAB 1

PENDAHULUAN

1.1 Deskripsi Masalah

Pada tugas ini, mahasiswa diminta membuat program yang mensimulasikan transformasi linier untuk melakukan operasi translasi, refleksi, dilatasi, rotasi, dan sebagainya pada sebuah objek 2D dan 3D. Objek dibuat dengan mendefinisikan sekumpulan titik sudut lalu membuat bidang 2D/3D dari titik-titik tersebut. Contoh objek 2D: segitiga, segiempat, polygon segi-n, lingkaran, rumah, gedung, mobil, komputer, lemari, dsb. Contoh objek 3D: kubus, pyramid, silinder, terompet, dll. Program akan memiliki dua buah window, window pertama (command prompt) berfungsi untuk menerima input dari user, sedangkan window kedua (GUI) berfungsi untuk menampilkan output berdasarkan input dari user. Kedua window ini muncul ketika user membuka file executable. Untuk objek 2D, saat program baru mulai dijalankan, program akan menerima input N, yaitu jumlah titik yang akan diterima. Berikutnya, program akan menerima input N buah titik tersebut (pasangan nilai x dan y). Setelah itu program akan menampilkan output sebuah bidang yang dibangkitkan dari titik-titik tersebut. Selain itu juga ditampilkan dua buah garis, yaitu sumbu x dan sumbu y. Nilai x dan y memiliki rentang minimal -500 pixel dan maksimum 500 pixel. Pastikan window GUI yang Anda buat memiliki ukuran yang cukup untuk menampilkan kedua sumbu dari ujung ke ujung. Hal yang sama juga berlaku untuk objek 3D tetapi dengan tiga sumbu: x, y, dan z.

Transformasi yang harus dilakukan :

1. Translasi
Melakukan translasi objek dengan menggeser nilai x sebesar dx dan menggeser nilai y sebesar dy.
2. Dilatasi
Melakukan dilatasi objek dengan faktor scaling k.
3. Rotasi
Melakukan rotasi objek secara berlawanan arah jarum jam sebesar deg derajat terhadap titik a,b
4. Refleksi
Melakukan pencerminan objek. Nilai param adalah salah satu dari nilai- nilai berikut: x, y, $y=x$, $y=-x$, atau (a,b). Nilai (a,b) adalah titik untuk melakukan pencerminan terhadap.
5. Shear
Melakukan operasi shear pada objek. Nilai param dapat berupa x (terhadap sumbu x) atau y (terhadap sumbu y). Nilai k adalah faktor shear.
6. Strecth

Melakukan operasi stretch pada objek. Nilai param dapat berupa x (terhadap sumbu x) atau y (terhadap sumbu y). Nilai k adalah faktor stretch.

7. Custom

Melakukan transformasi linier pada objek dengan matriks transformasi sesuai dengan masukan pengguna

8. Multiple

Melakukan transformasi linier pada objek sebanyak n kali berurutan. Setiap baris input $1..n$ dapat berupa translate, rotate, shear, dll tetapi bukan multiple, reset, exit.

9. Reset

Mengembalikan objek pada kondisi awal objek didefinisikan.

BAB 2

TEORI SINGKAT

2.1 Bahasa Python

Python merupakan salah satu bahasa pemrograman yang banyak digunakan saat ini. Python memiliki syntax yang sederhana. Python dapat digunakan untuk membuat program dengan paradigma prosedural, fungsional, maupun pemrograman berorientasi objek. Bahasa python menerjemahkan program yang ditulis dengan dieksekusi per baris (interpreter). Python menyediakan berbagai library yang *powerful* salah satunya adalah library numpy yang mempermudah programmer dalam mengelola array dan array multidimensi (matriks).

2.2 OpenGL GUI

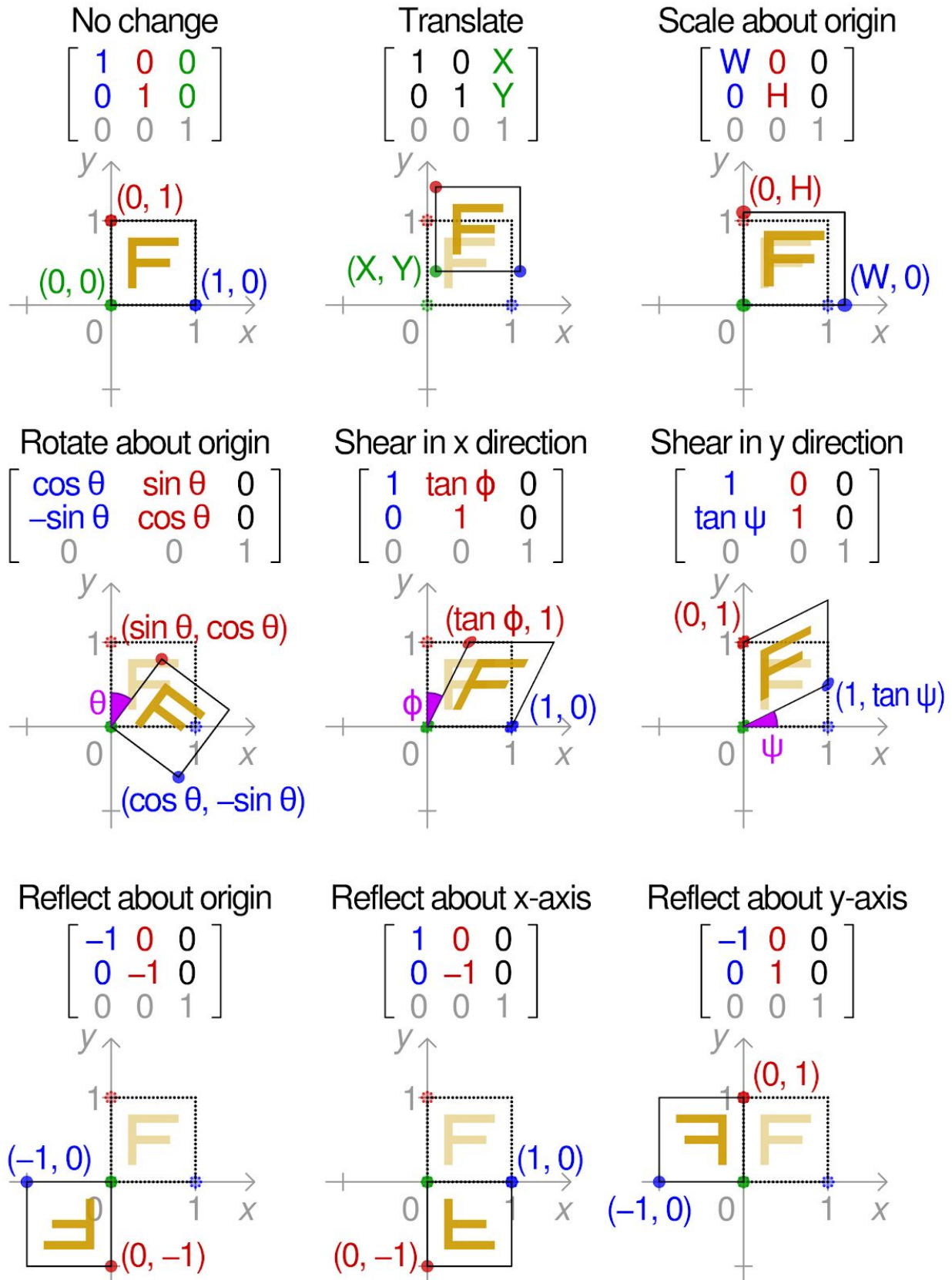
OpenGL (Open Graphics Library) adalah suatu aplikasi yang dapat digunakan bersama bahasa pemrograman yang lain untuk menghasilkan sebuah objek grafis dan mengolahnya. Objek grafis tersebut dapat digambarkan pada jendela OpenGL. OpenGL dapat digunakan untuk membuat objek 1 dimensi, 2 dimensi, 3 dimensi.

2.4 pygame

Pygame adalah modul lintas platform python yang digunakan dalam pembuatan sebuah game. Selain itu pygame juga dapat digunakan untuk mensimulasikan transformasi geometri pada sebuah benda. Modul pygame dapat digunakan oleh program python dengan menambahkannya (import) sebagai library program.

2.3 Transformasi geometri

Transformasi geometri merupakan suatu proses perubahan bentuk objek, baik objek 2D maupun objek 3D. Transformasi geometri diimplementasikan dengan memodelkan objek yang akan di transformasikan dengan titik titik yang merupakan titik sudut bidang. Kemudian titik titik tersebut di susun dalam sebuah matriks. Setelah itu matriks tersebut dikalikan dengan sebuah matriks yang bersesuaian dengan transformasi yang dilakukan. Beberapa jenis transformasi geometri dasar adalah translasi, dilatasi, rotasi, refleksi, shear, stretch.



Gambar 2.1 Matriks matriks transformasi dan visualisasinya

BAB 3

IMPLEMENTASI PROGRAM

3.1 Garis Besar Program

Program terdiri dari sebuah program utama yang berfungsi sebagai program yang akan menampilkan menu, menerima input fungsi dan juga menggambar bangun datar dan bangun ruang menggunakan pygame dan OpenGL, program utama akan meng-include 2 buah file yang masing-masing berisi fungsi-fungsi untuk melakukan transformasi 2 dimensi dan 3 dimensi. Saat program utama dibuka maka program akan terbuka di terminal/command prompt untuk keperluan input selain itu juga window dari opengl dan pygame akan muncul untuk memproyeksikan bangun yang akan dioperasikan.

Saat menggunakan program utama, user akan disuruh untuk memilih mode transformasi yakni 2D atau 3D. Jika memilih pilihan 2D selanjutnya user akan disuruh untuk menginput bangun datar yang akan digunakan yakni berupa input jumlah titik sudut dilanjutkan koordinat titik-titik dalam bidang x,y , kemudian program akan menggambar polygon sesuai input. Selanjutnya user dapat memasukkan input fungsi transformasi yang tersedia dengan parameternya masing-masing dalam bentuk string dan program akan menampilkan hasil transformasi ke layar. Hal yang sama juga dilakukan pada pilihan mode 3D hanya saja bangun ruang yang tersedia hanyalah berupa kubus untuk mempermudah penggambaran, untuk fungsi 3D yang tersedia sama seperti pada transformasi 2D hanya berbeda pada parameternya saja.

3.2 Definisi Struktur

Program terdiri dari 3 modul / unit yaitu :

1. Render

Bagian ini berisi program untuk menampilkan objek ke pop up window. Pada transformasi 2D digunakan pyOpenGL sebagai window. Sedangkan pada transformasi 3D digunakan pygame sebagai window. Proses menggambar objek 2D. Dimulai dengan membaca input dari pengguna berupa banyak titik (n). Kemudian program meminta n buah titik (x,y) sebagai inputan. Setelah itu program akan menyimpan input tersebut dalam sebuah matriks M . Setelah itu program akan menggambar objek 2D dengan menghubungkan titik titik yang ada dalam matriks M . Pada objek 3D kami memilih kubus sebagai objek transformasi. Kami mendefinisikan kubus dengan 8 titik sudut dan 12 sisi yang dibuat di program render. Kemudian titik titik sudut disimpan dalam matriks. Saat program dijalankan pygame akan menggambar kubus berdasarkan sisi dan titik yang tersimpan dalam matriks.

Pada bagian render penulis menambahkan fungsi untuk scale up layar dan scale down layar. Sehingga objek dapat dilihat walau dalam koordinat yang jauh dari titik origin.

Pada render 3D program juga didesain untuk melihat objek dengan berbagai *angle* dengan memutar sumbu x, y, dan z.

2. MatrixOperation

Penulis membagi bagian ini menjadi dua bagian yaitu : operasi matriks transformasi 2D dan operasi matriks transformasi 3D. Matriks transformasi pada 2D berukuran $n \times 3$, n adalah banyaknya titik yang membangun objek, 3 merupakan banyaknya kolom yang merepresentasikan transformasi pada koordinat x, y, dan kolom terakhir diisi sesuai transformasi yang bersesuaian. Matriks transformasi 2D dibentuk berdasarkan transformasi yang dilakukan. Sehingga penulis mendefinisikan matriks transformasi yang berbeda beda sejumlah transformasi yang akan dilakukan. Sedangkan pada matriks 3D terdapat 4 kolom yang merepresentasikan koordinat x, y, z, dan kolom terakhir diisi sesuai transformasi yang dilakukan.

Operasi matriks dilakukan dengan mengalikan matriks yang berisi titik titik awal dengan matriks transformasi. Kemudian diperoleh matriks baru yang merupakan hasil transformasi yang bersesuaian. Untuk merealisasikan hal tersebut dibuat suatu fungsi perkalian matriks yang menambah keefektifan program.

3. Main

Pada bagian ini program mengintegrasikan render dan matrix operation menjadi satu kesatuan. Bagian ini juga berisi interaksi input output antara pengguna dan program.

3.3 Pembagian Tugas

Dalam pengerjaan tugas besar ini awalnya penulis melakukan diskusi untuk menentukan algoritma dan program yang akan diimplementasikan. Penulis juga menggunakan github dalam memudahkan pengerjaan tugas. Penulis berkomunikasi satu sama lain melalui aplikasi chatting sehingga pengerjaan tugas besar tetap dilakukan meski tidak berkumpul dalam satu tempat.

Pembagian tugas yang kami lakukan sebagai berikut :

1. Render : irfan
2. Matriks operation : zhafran, fajar, irfan
3. Main dan debugging : zhafran ,irfan,fajar
4. Fungsi kalimatriks : fajar, irfan
5. Scale up layar : zhafran, fajar
6. Pembuatan laporan : irfan, fajar
7. Merapikan program : irfan, zhafran

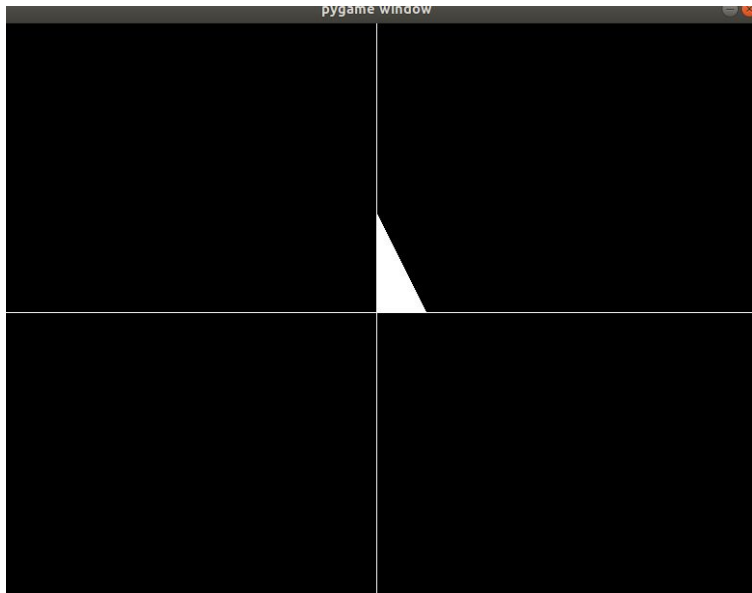
BAB 4

EKSPERIMEN

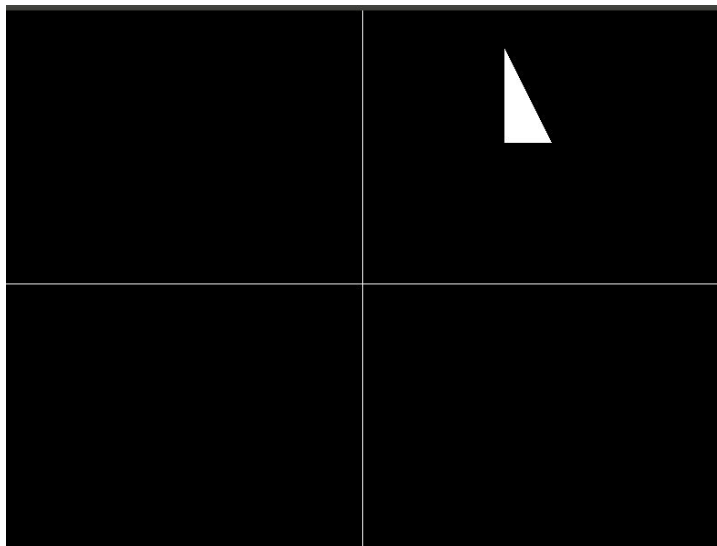
4.1 Hasil Eksekusi Program

A. 2D

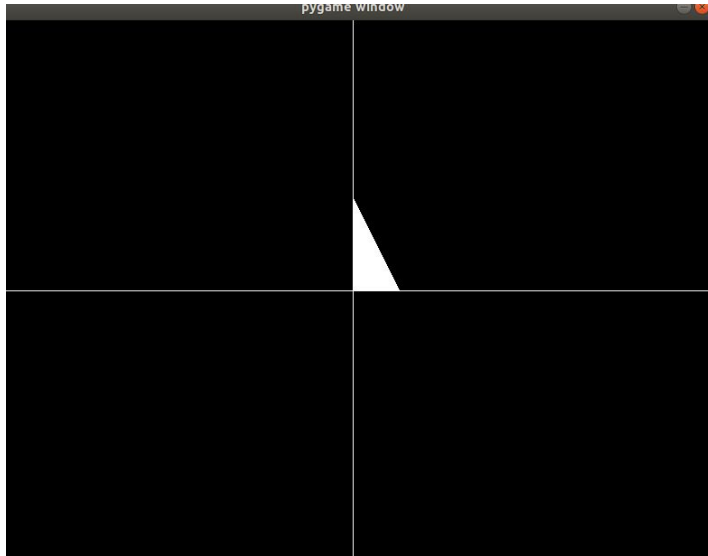
1. Translate ($dx = 3$, $dy = 3$)
sebelum



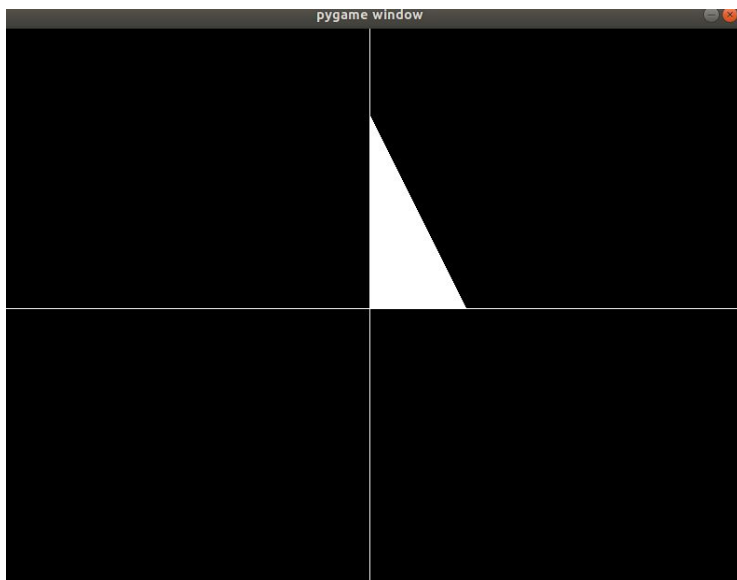
Sesudah



2. Dilatasi dengan faktor 2
Sebelum

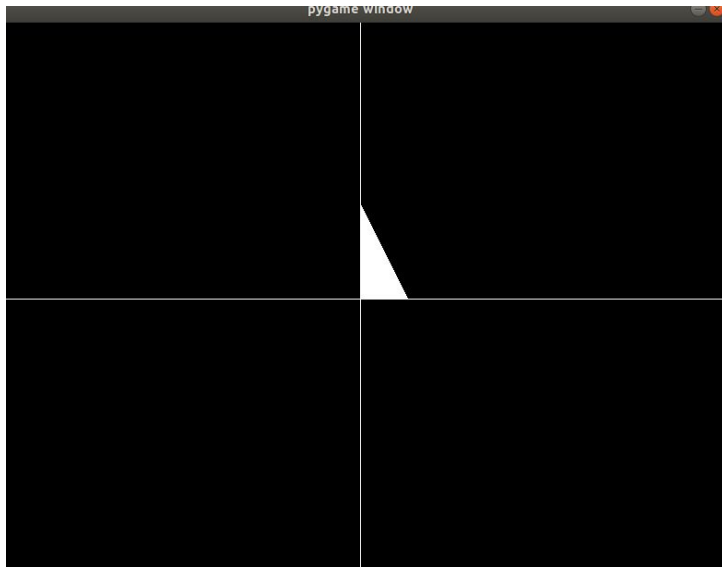


Sesudah

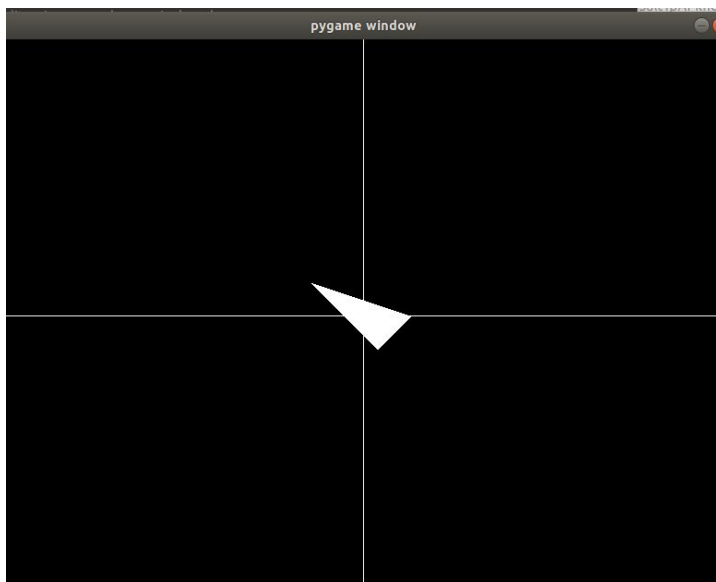


3. Rotasi(45 derajat terhadap sumbu x)

Sebelum

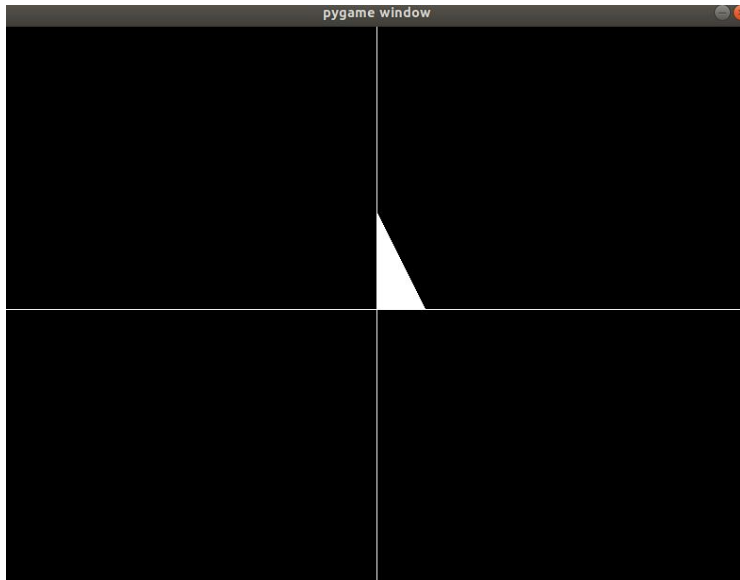


Sesudah

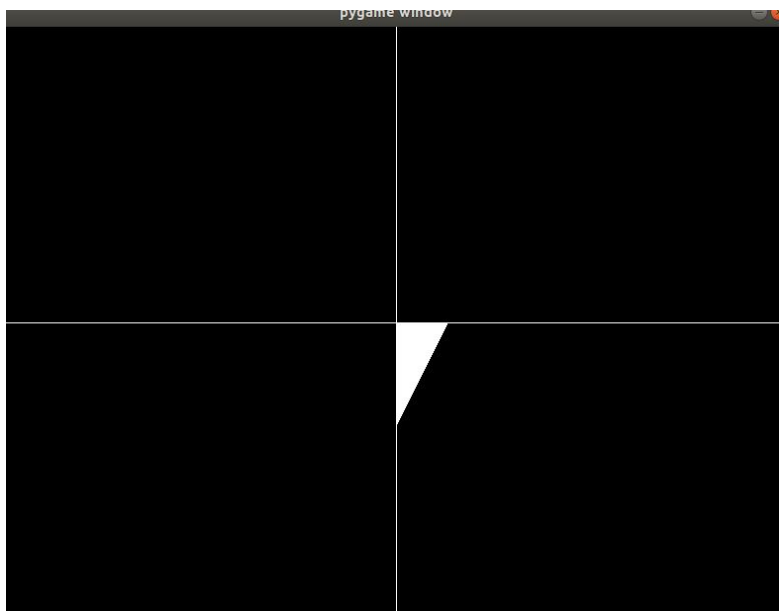


4. Refleksi(refleksi terhadap sumbu x)

Sebelum

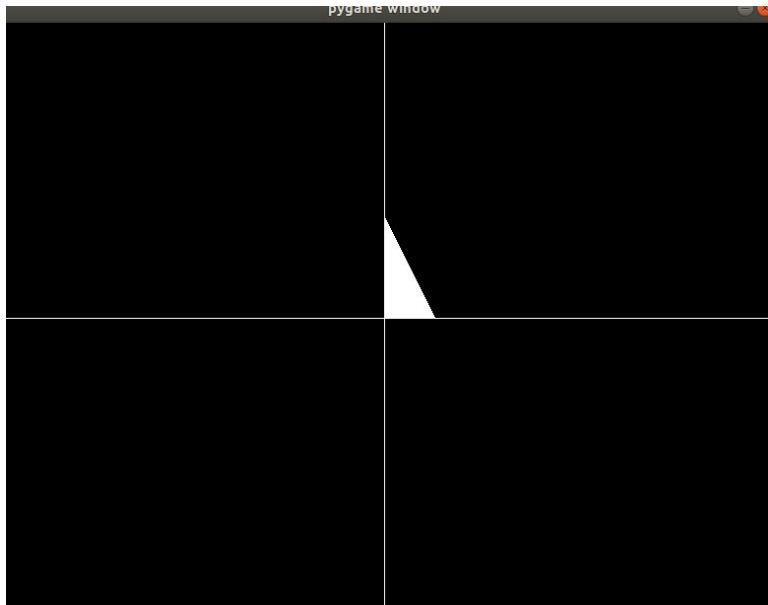


Sesudah

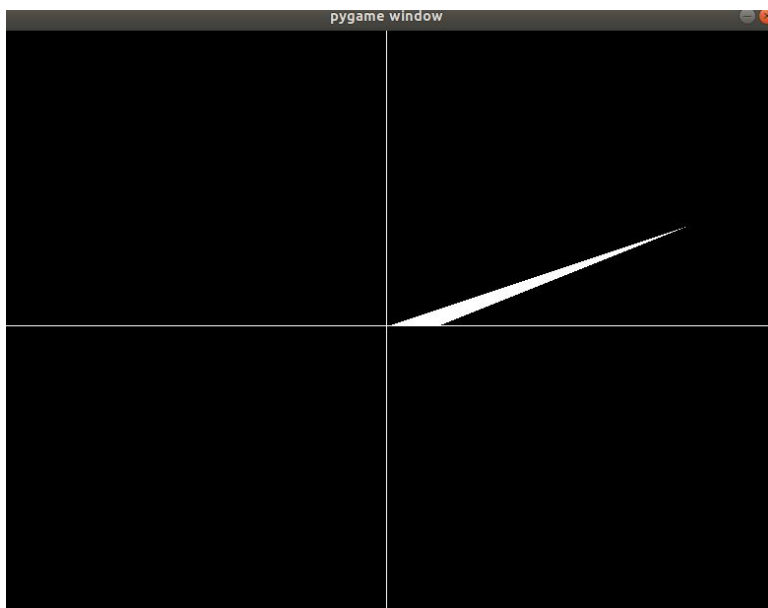


5. Shear (shear x sebesar 3 satuan)

Sebelum

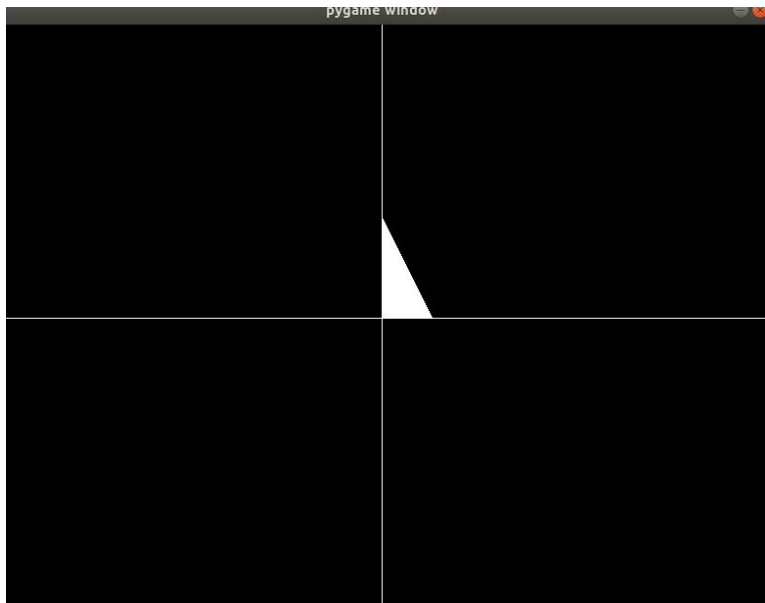


Sesudah

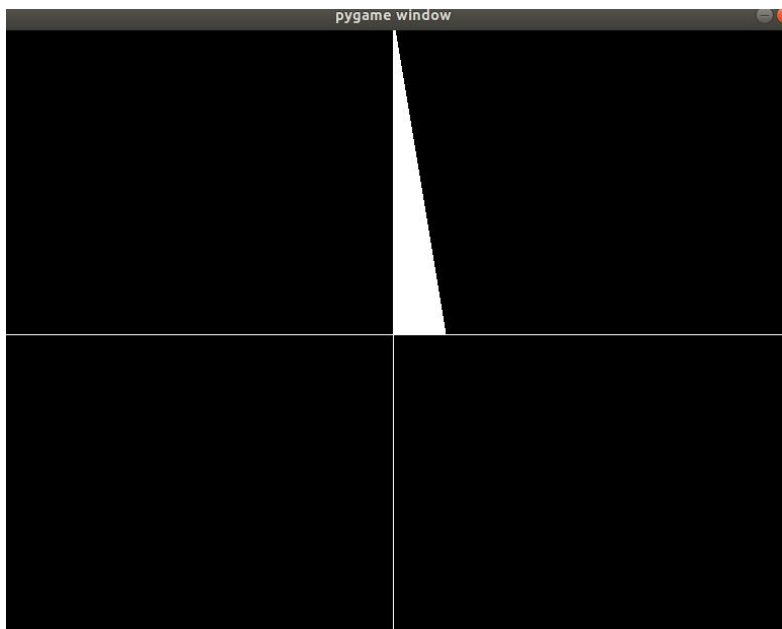


6. Stretch (stretch y sebesar 3 satuan)

Sebelum

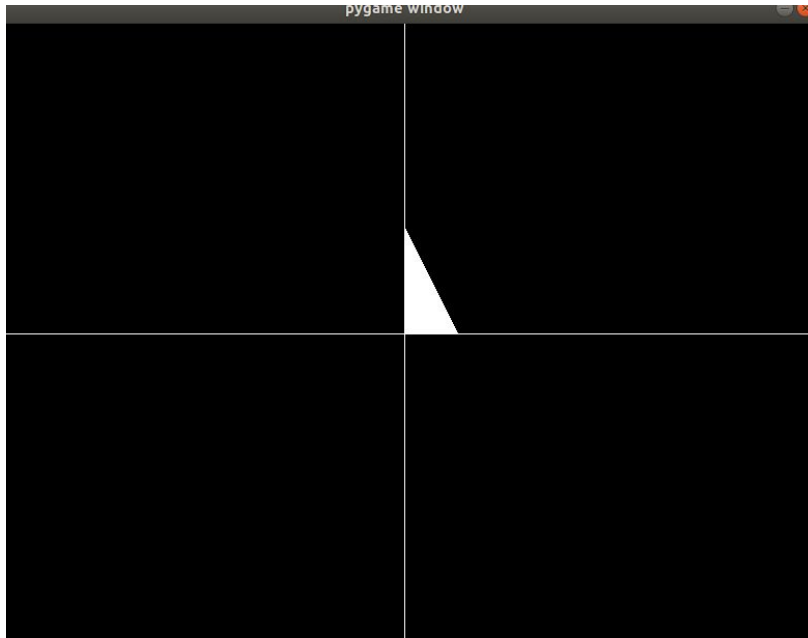


Sesudah

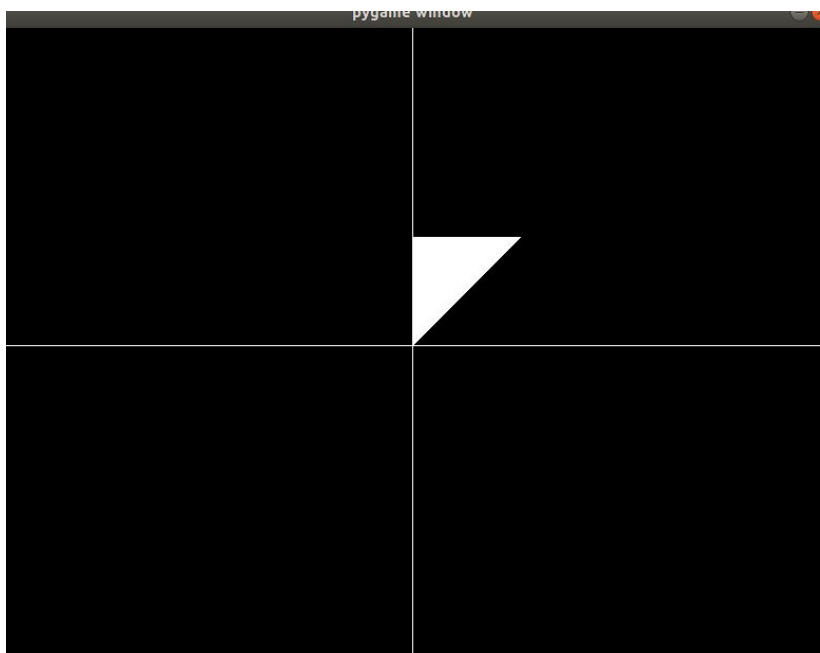


7. Custom (matriksnya $\begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix}$)

Sebelum



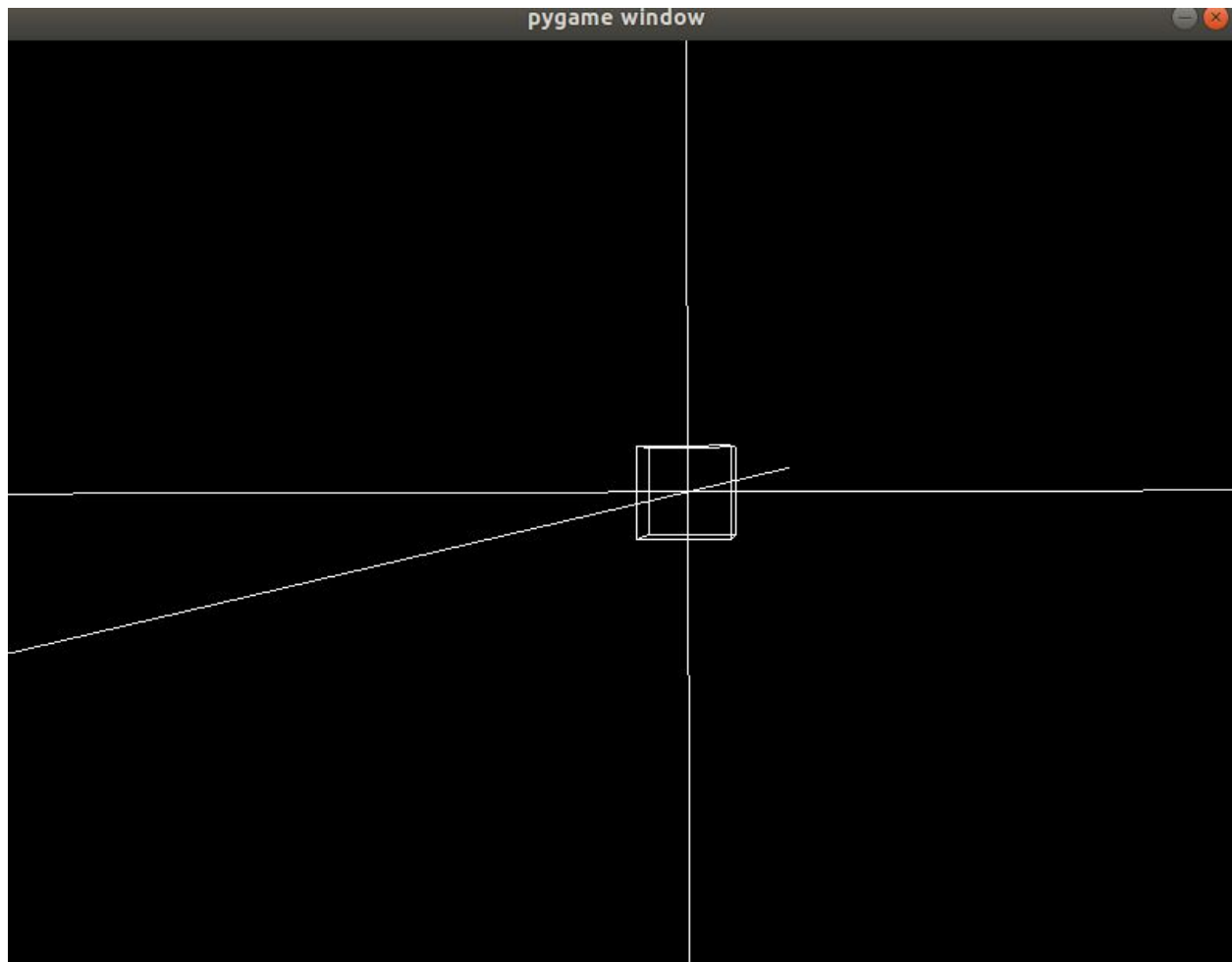
Sesudah



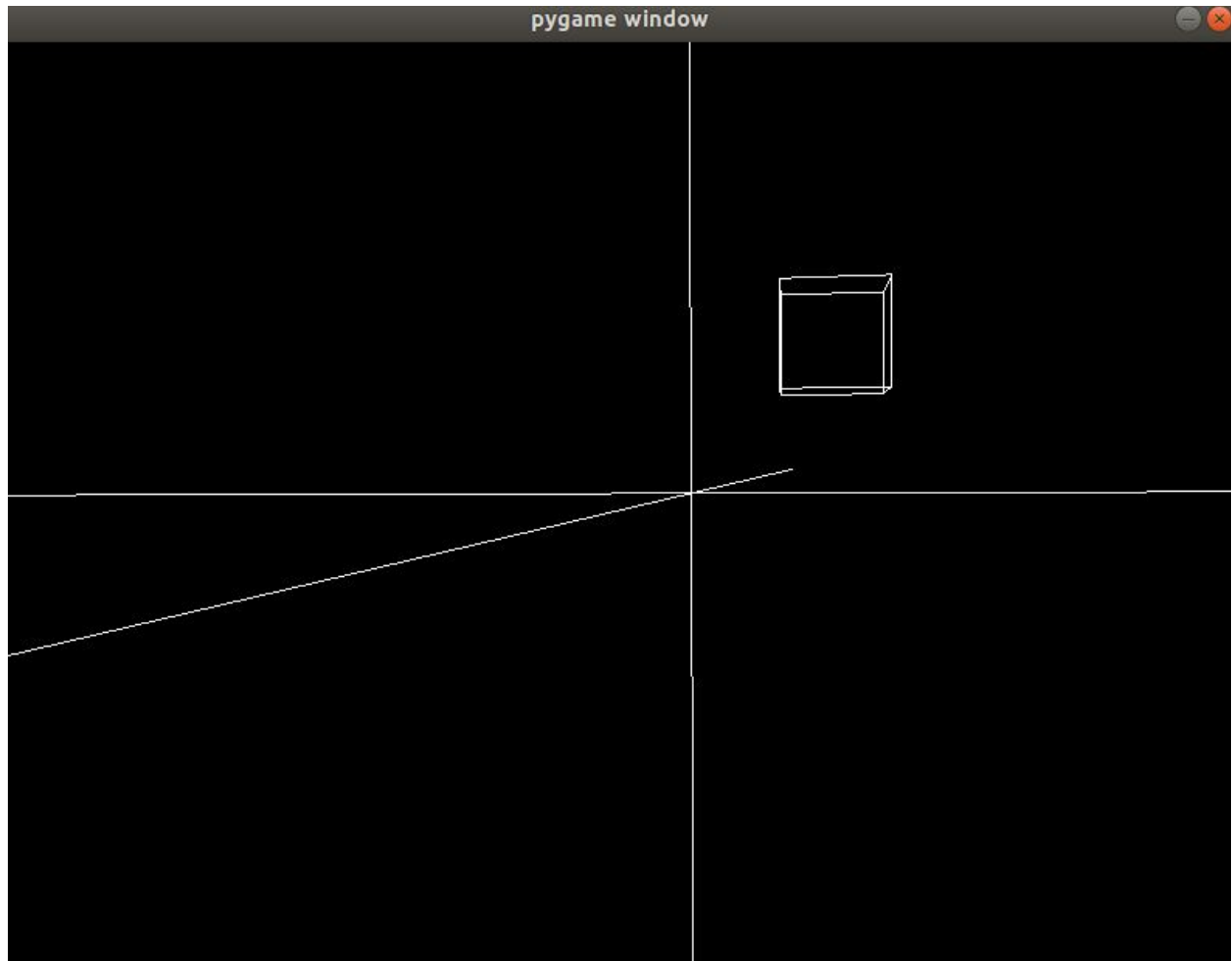
B. 3D

1. Translasi (translasi $dx = 3$, $dy = 3$, $dz = 3$)

Sebelum

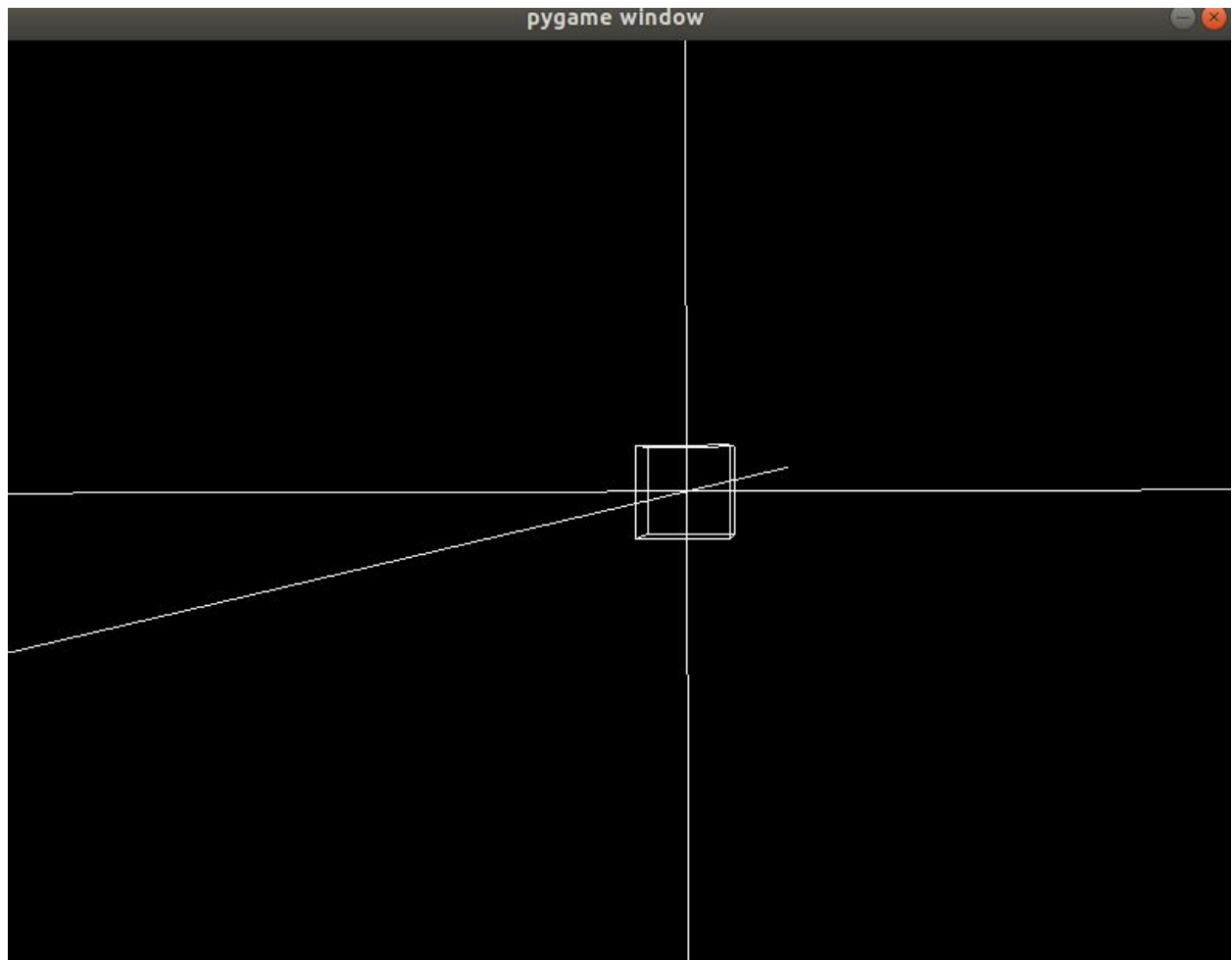


Sesudah

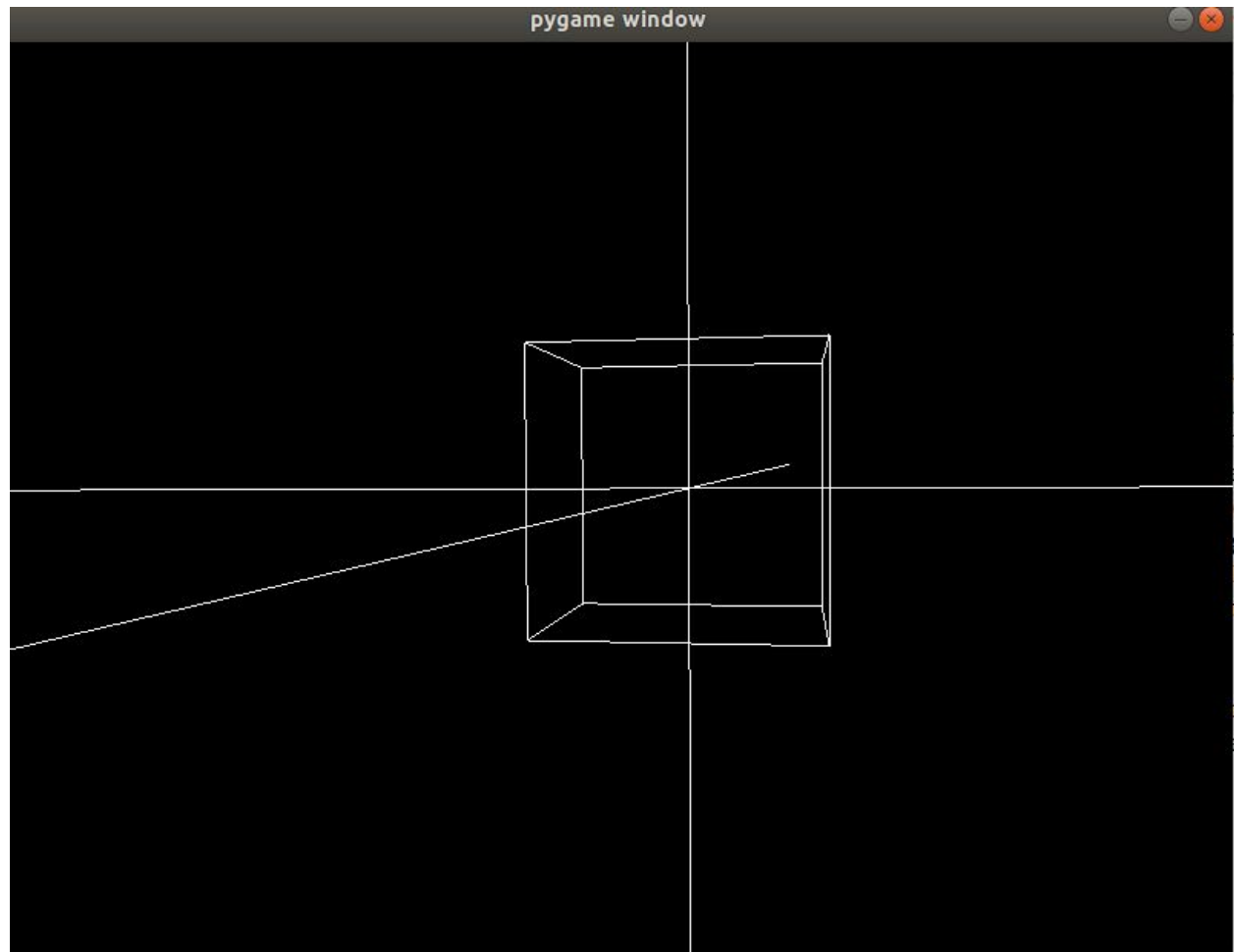


2. Dilatasi (dilatasi dengan faktor $k = 3$)

Sebelum

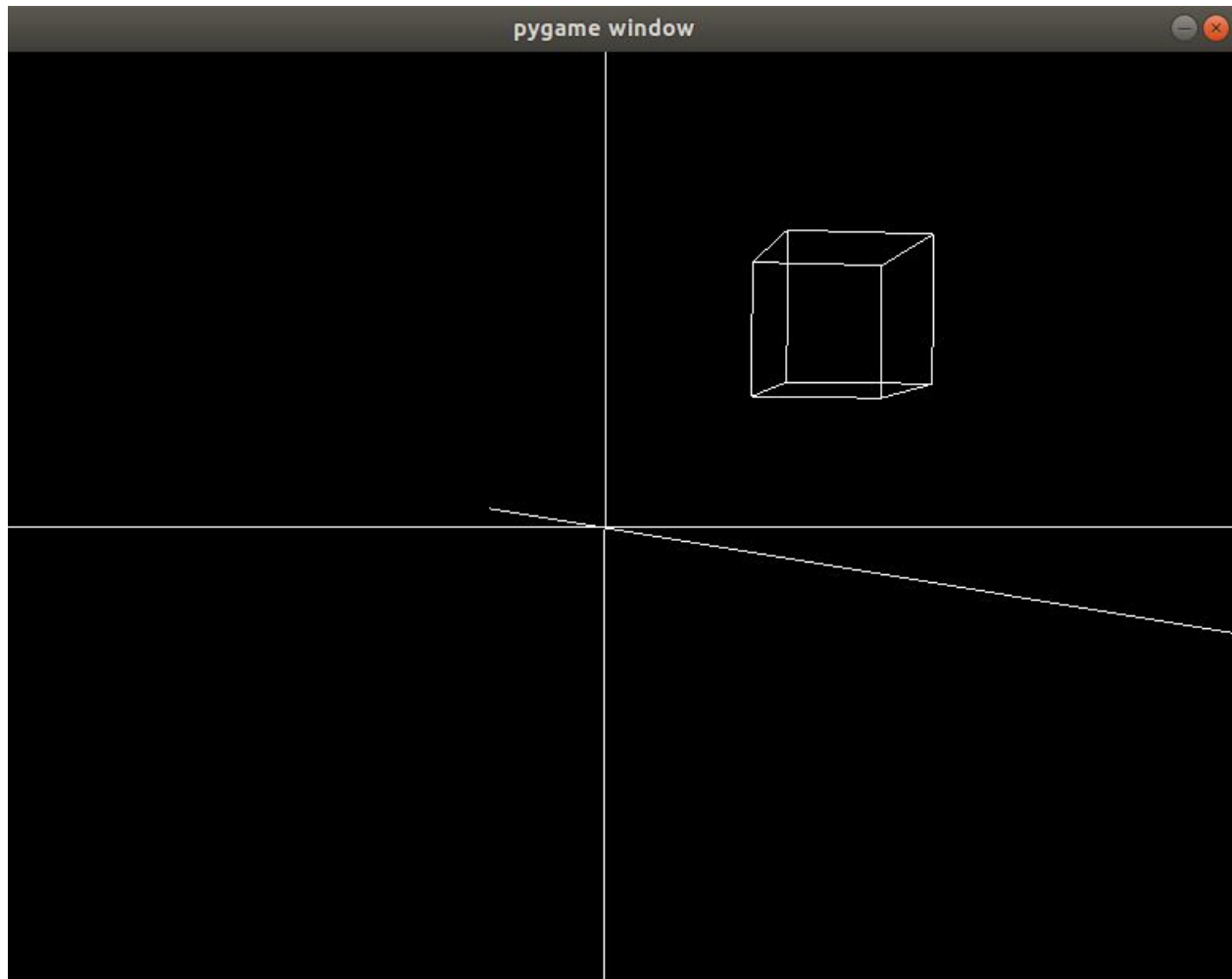


Sesudah

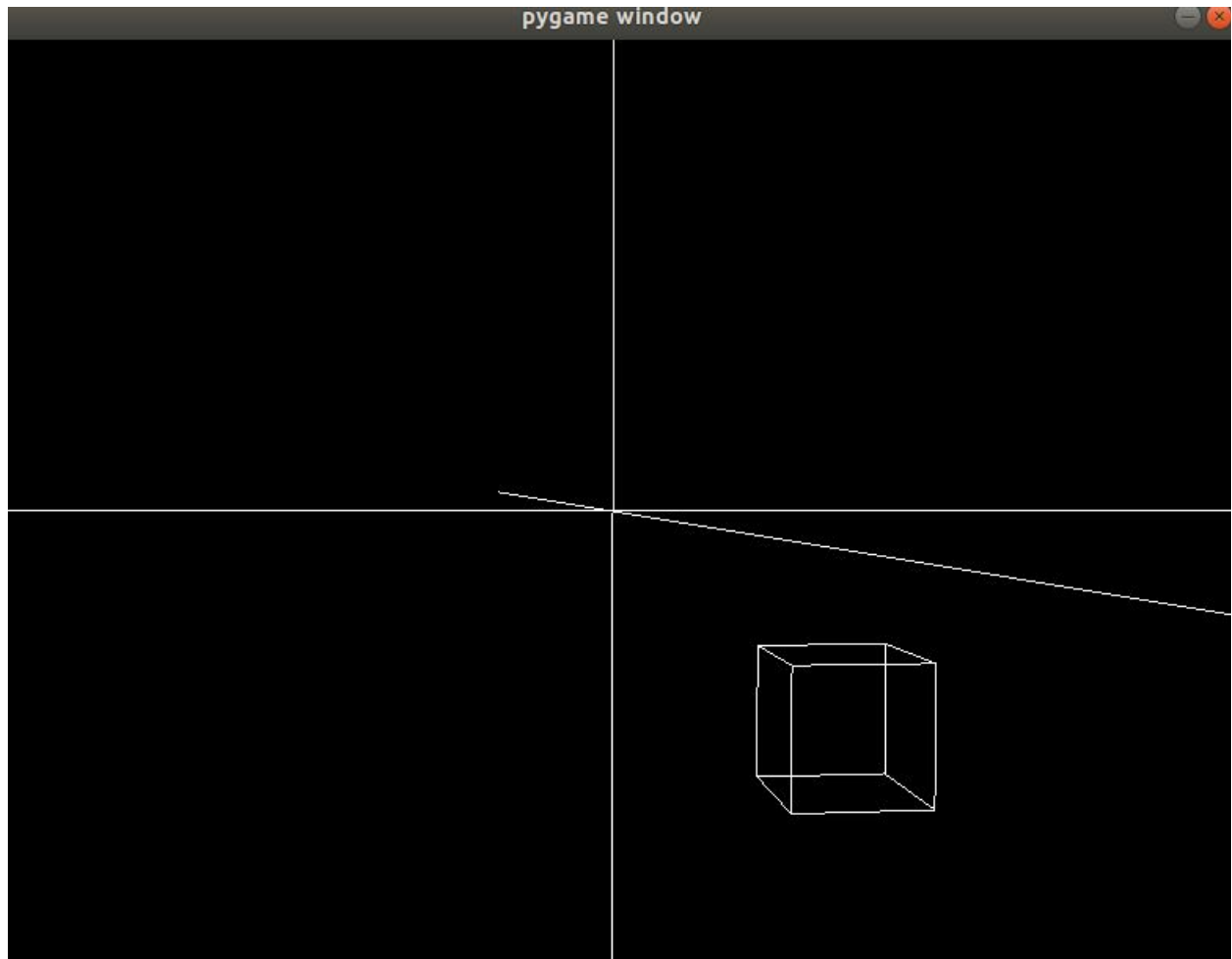


3. Refleksi

Sebelum

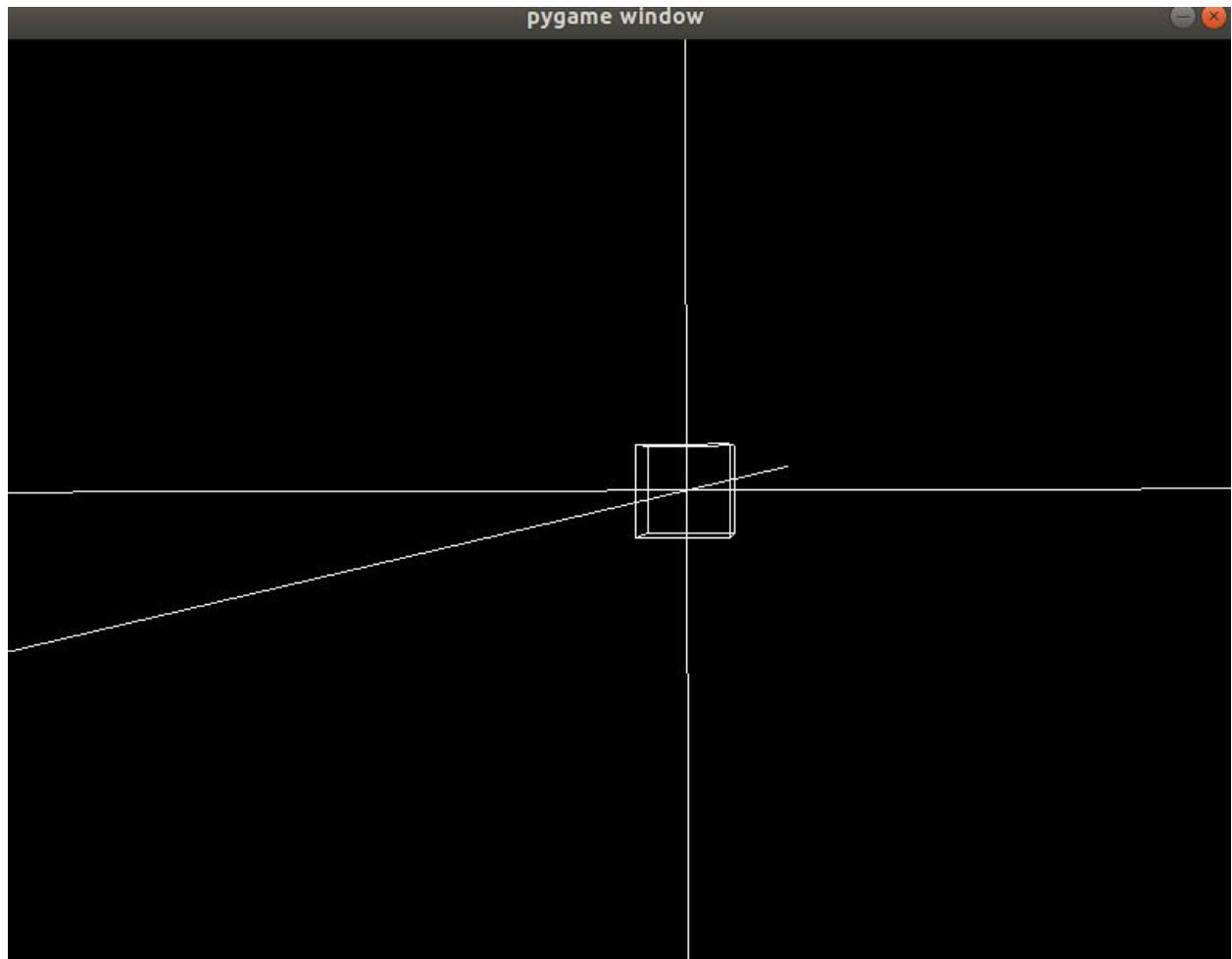


Sesudah

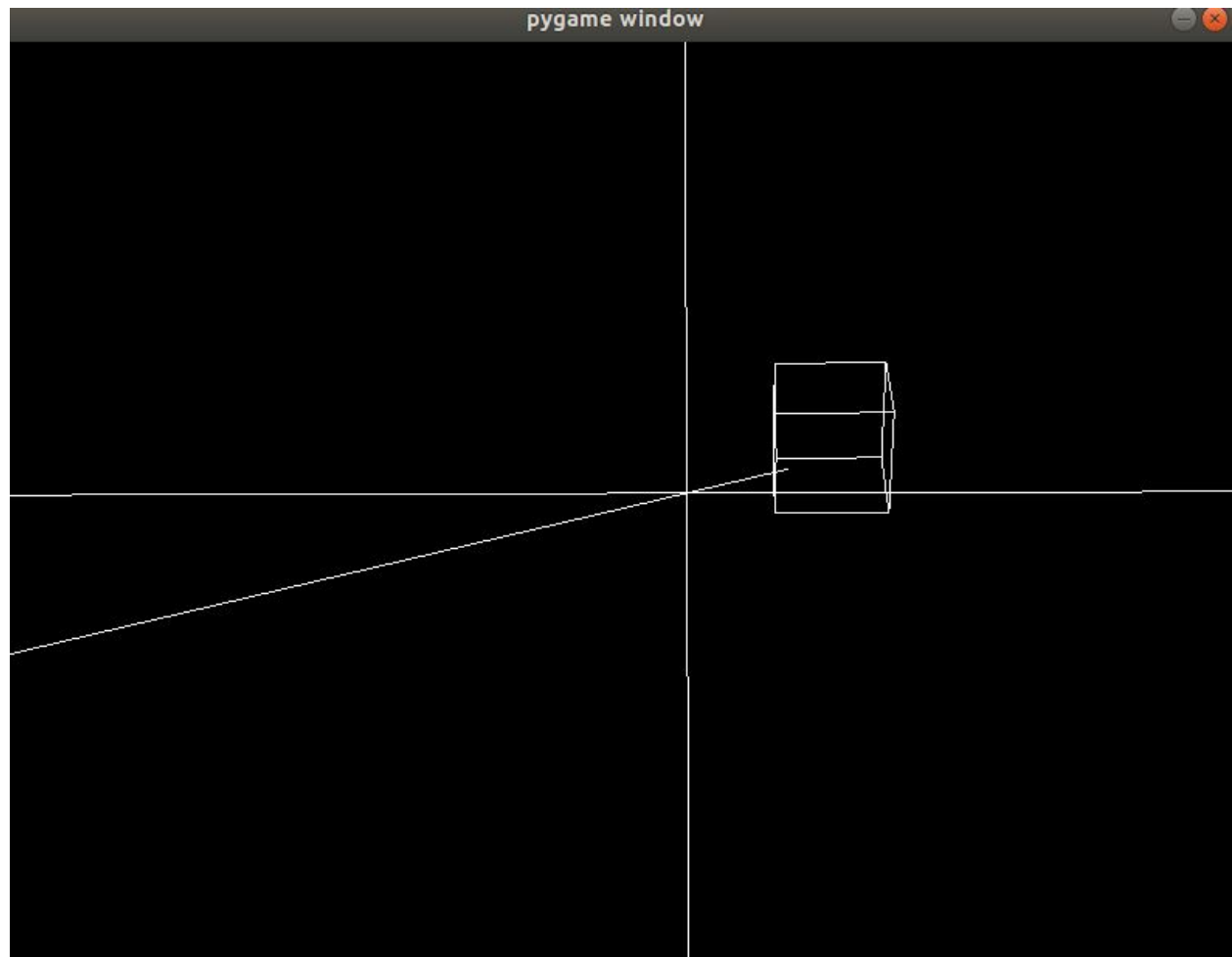


4. Rotasi (30 derajat terhadap sumbu x)

Sebelum

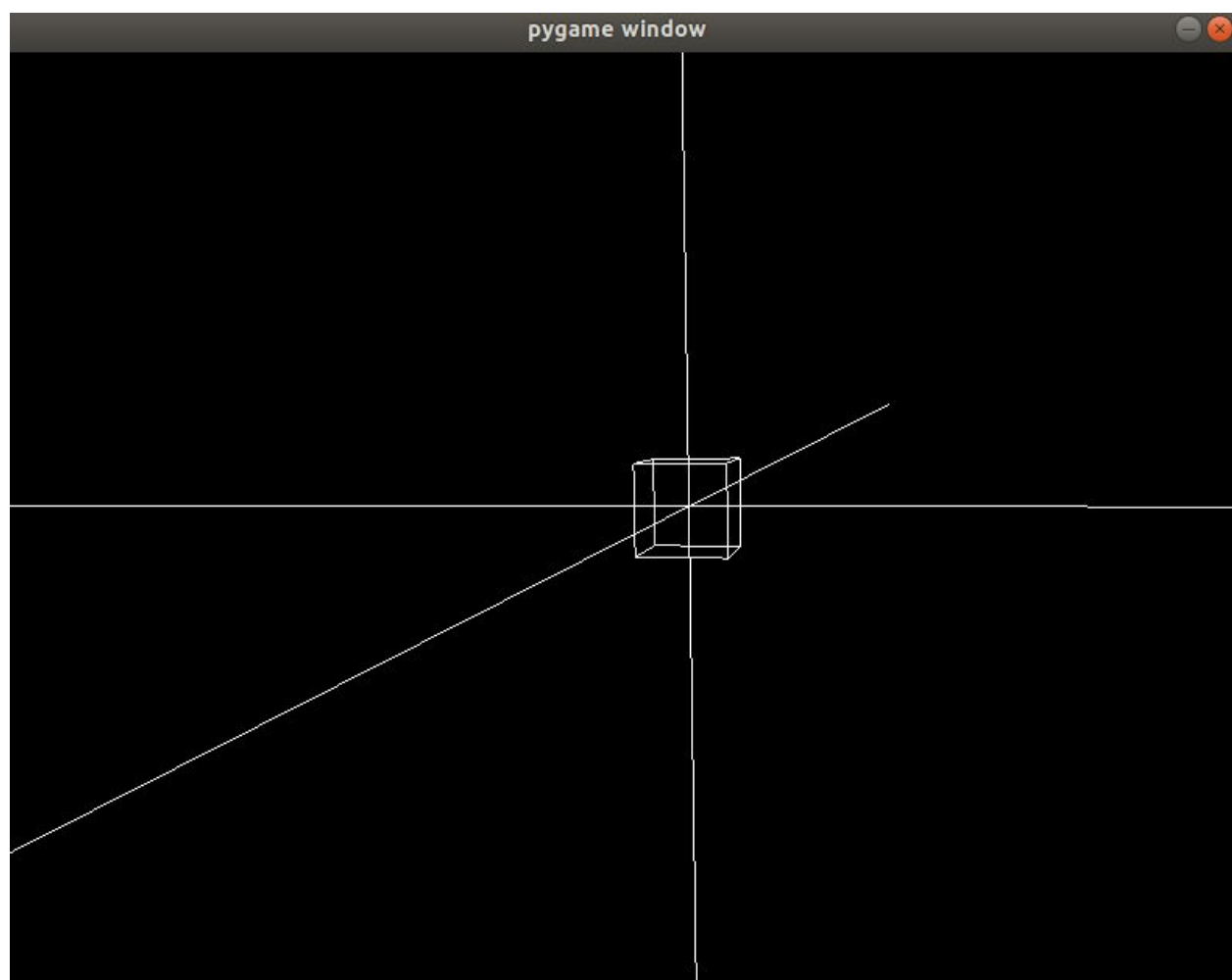


Sesudah

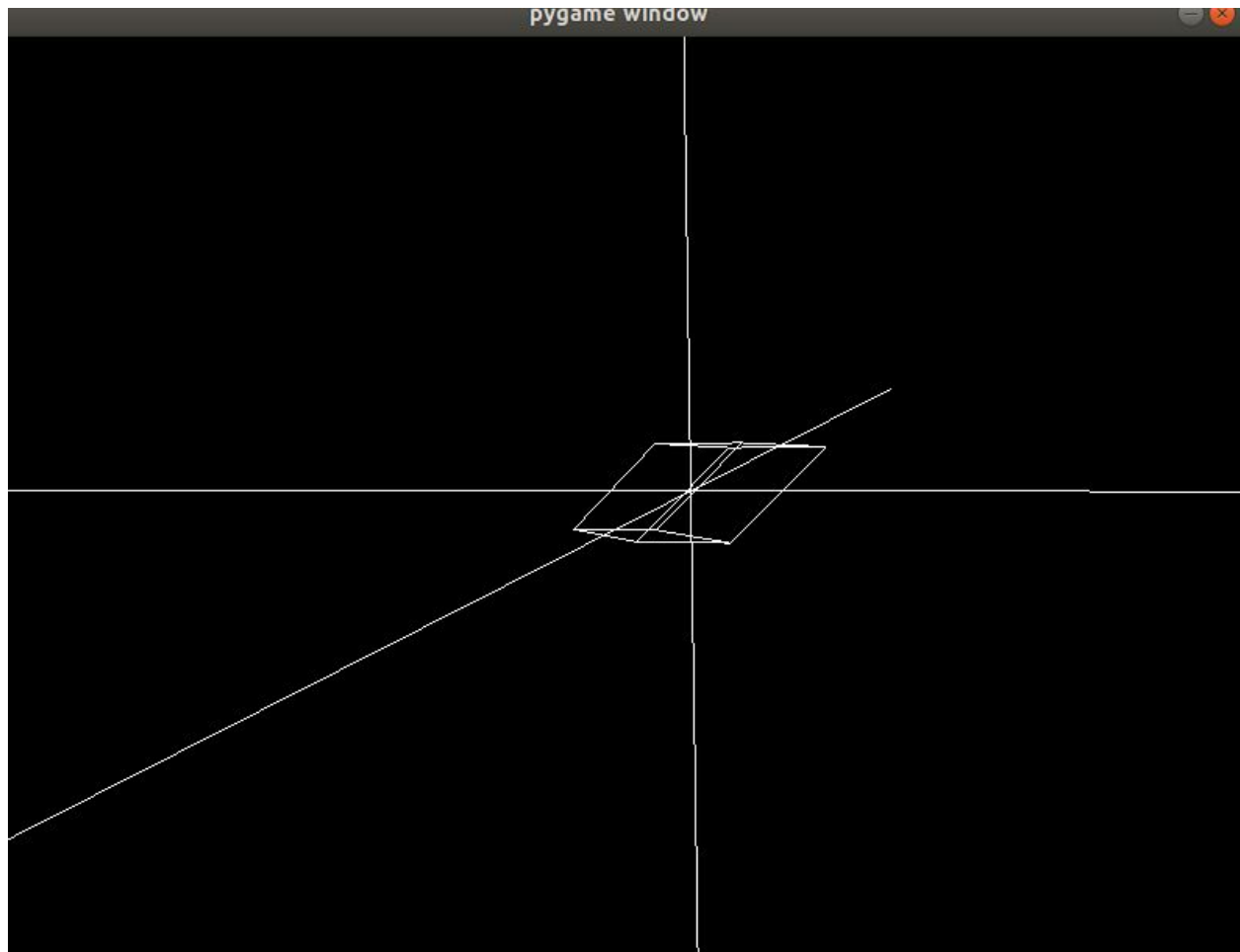


5. Shear (x dengan $k = 1$)

Sebelum

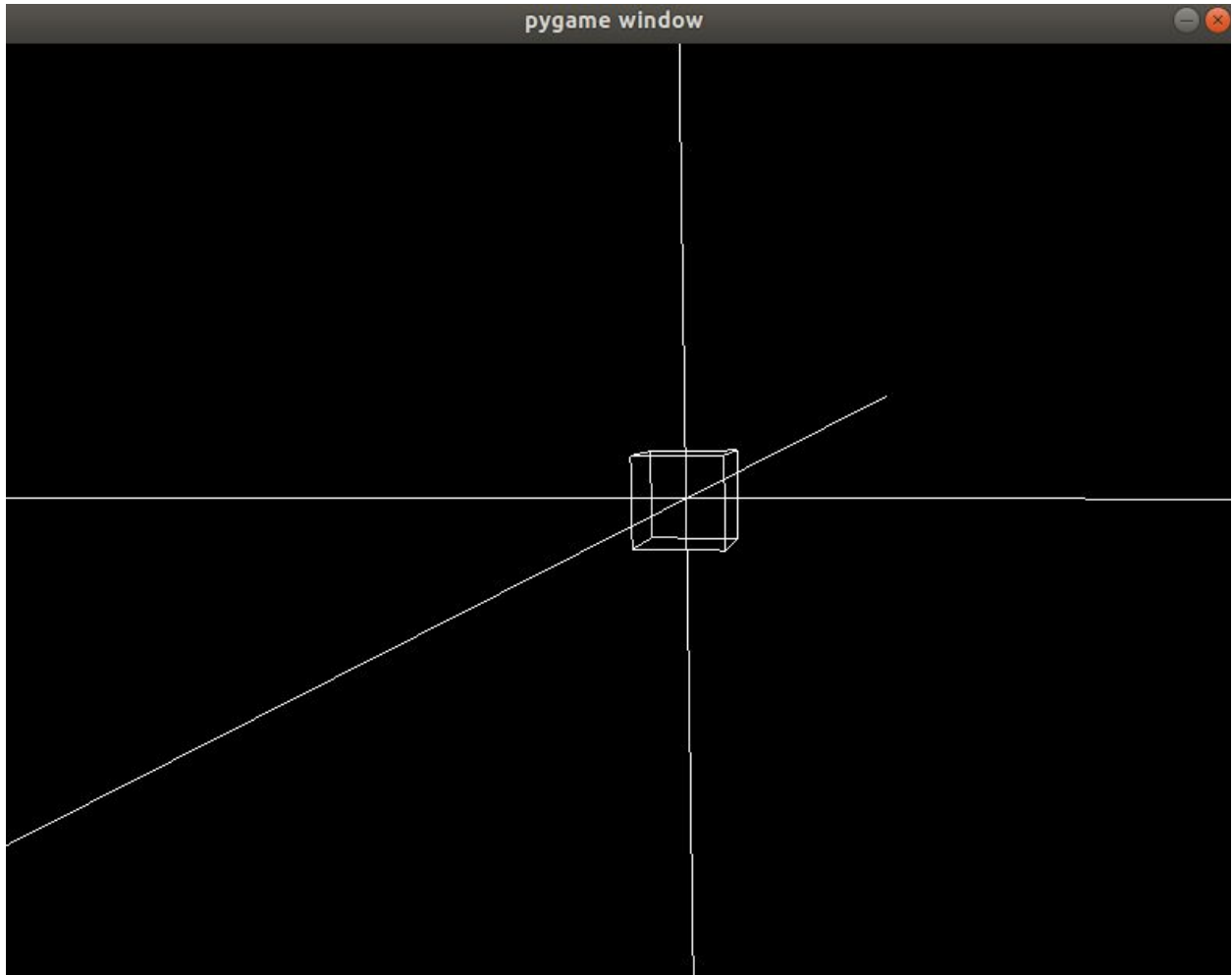


Sesudah

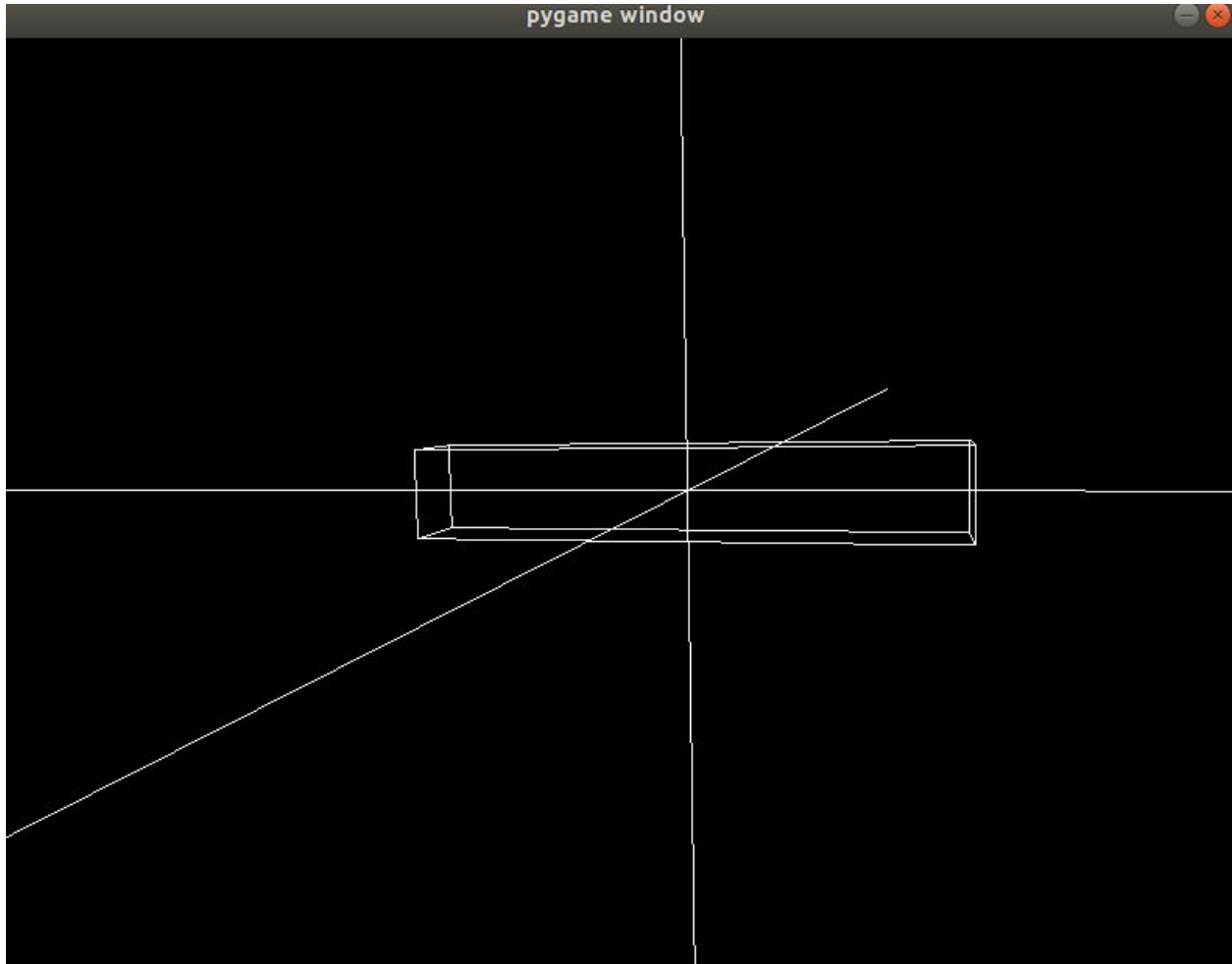


6. Stretch ($k = 6$)

Sebelum

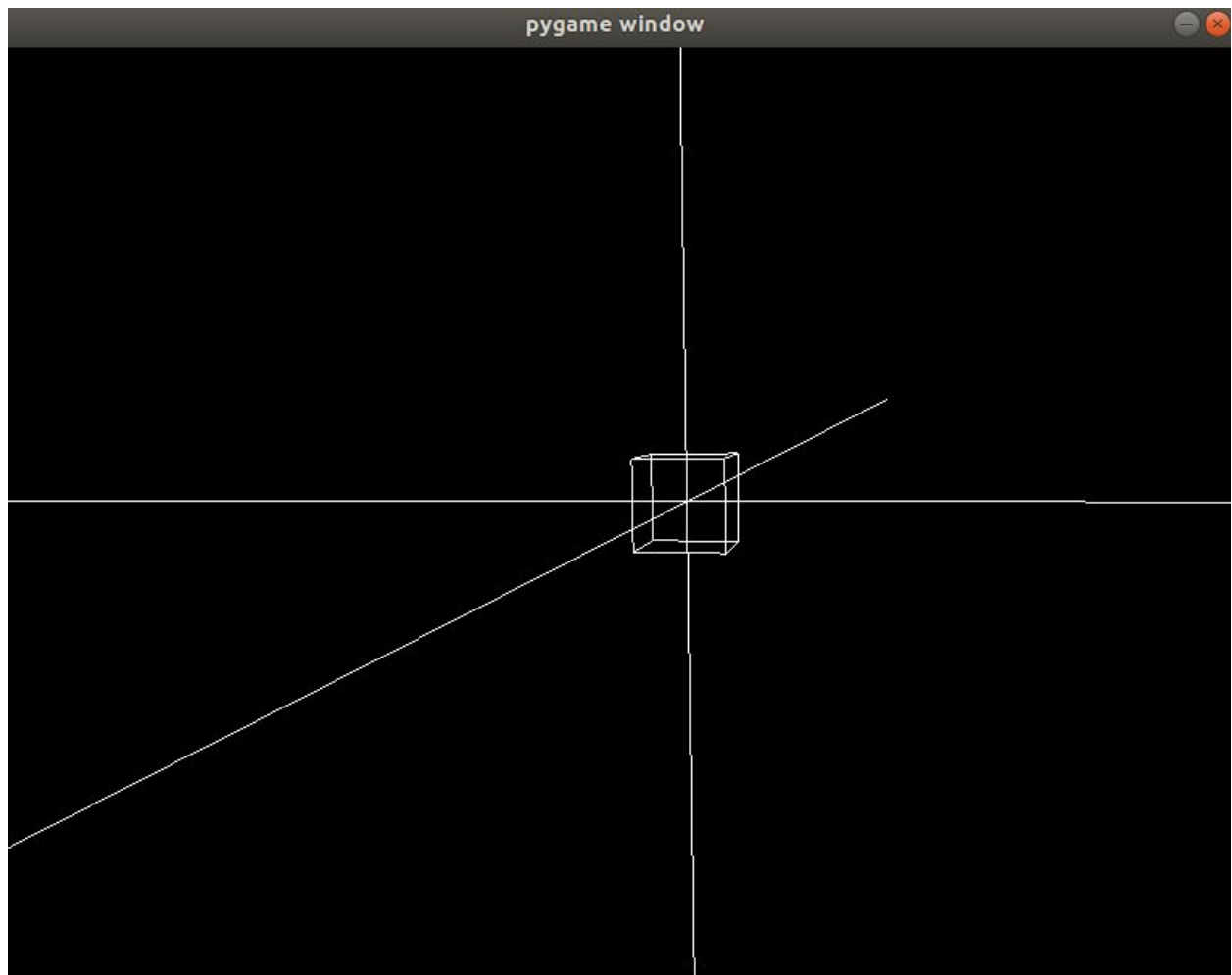


Sesudah

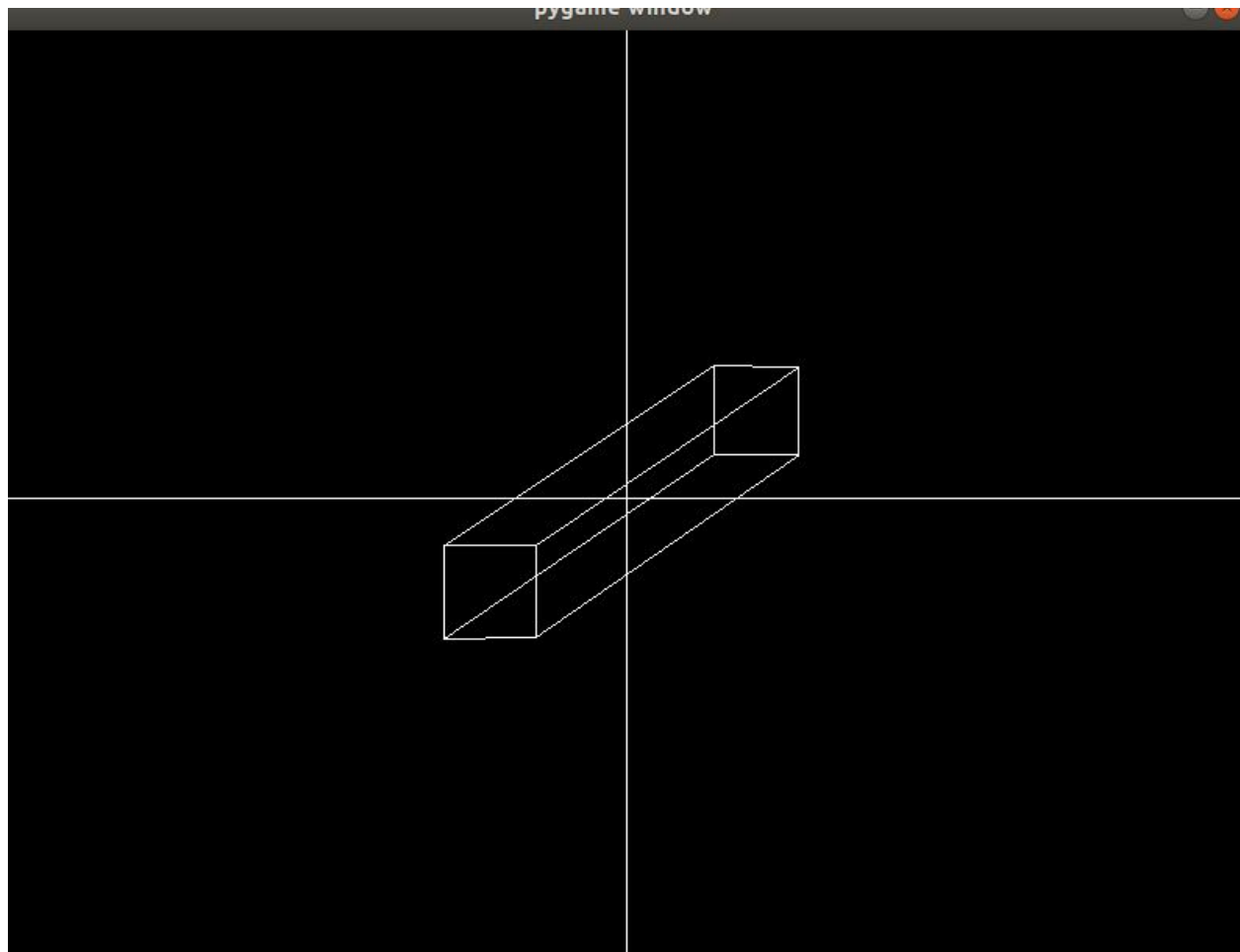


7. Custom (dengan matriks $\begin{bmatrix} 1 & 0 & 3 \\ 0 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$)

Sebelum



Sesudah



4.2 Analisis

Penulis menyusun program dengan terlebih dahulu mempelajari bahasa python, library pyOpenGL, dan library pygame. Kemudian penulis menggambarkan objek 2D dan 3D dengan bahasa python dan bantuan library. Setelah itu penulis membuat suatu matriks transformasi pada berbagai transformasi geometri. Setelah itu objek akhir hasil transformasi merupakan perkalian dari matriks objek sebelum transformasi dengan matriks transformasi yang bersesuaian.

Pada penyusunan matriks 3D, penulis mempelajari matriks transformasi 2D, kemudian menyesuaikan matriks tersebut dengan bentuk 3D. Konsep aljabar geometri dan transformasi geometri kami terapkan dalam menyusun matriks dan algoritma program.

Setiap program dijalankan penulis menyimpan kondisi awal objek pada matriks. Hal ini bertujuan agar ketika perintah reset dijalankan program dapat mengembalikan objek yang sesuai.

Sedangkan ketika perintah multiple n dijalankan program melakukan transformasi setiap kali dalam masukan pengguna dari 1 sampai n.

Pada pengerjaan tugas besar ini penulis mengalami beberapa kesulitan yang ditemui. Diantaranya indentasi pada python. Kesulitan menemukan sumber belajar yang mudah dipahami, serta adanya bug pada program. Masalah masalah tersebut kami selesaikan bersama sama dengan berkoordinasi melalui bertemu secara langsung ataupun melalui chat

BAB 5

PENUTUP

5.1 Kesimpulan

1. Transformasi geometri dapat dilakukan pada objek 2D dan 3D menggunakan bahasa program python dan OpenGL
2. Transformasi geometri dapat dimodelkan sebagai perkalian matriks.
3. Transformasi geometri dapat diterapkan untuk memperoleh objek yang berbeda maupun mengkombinasikan objek objek yang tersedia

5.2 Saran

Diselenggarakan asistensi untuk membantu pengerjaan tugas besar dan bertanya mengenai spesifikasi tugas besar. Dikarenakan masih banyak spesifikasi tugas yang kurang jelas terutama pada bagian transformasi di bidang 3 dimensi yang disuruh untuk mengikuti dari bagian transformasi 2 dimensi, dalam hal ini akan muncul banyak kemungkinan dari transformasi 3 dimensi sehingga diperlukan batasan yang jelas dalam pengerjaan tugas.

5.3 Refleksi

Dalam pengerjaan tugas dan program ini tentu masih banyak kekurangan dan kesalahan. Meskipun tim pembuat menginginkan kesempurnaan dalam penyusunan program dan laporan ini akan tetapi pada kenyataannya masih banyak kekurangan yang perlu kami perbaiki. Hal ini dikarenakan masih minimnya pengetahuan penulis. Oleh karena itu kritik dan saran yang membangun dari dosen, tim asisten, pengguna yang mencoba program tersebut, dan pembaca laporan ini diharapkan sebagai bahan evaluasi untuk kedepannya.