

Tugas Besar II IF2123 Aljabar Geometri

Simulasi Transformasi Linier pada Bidang 2D dan 3D dengan Menggunakan OpenGL API

Batas pengumpulan : Senin, 19 November 2018, Pukul 8.00.

Arsip pengumpulan : - Laporan (*hardcopy* dan *softcopy*)

- Soft copy folder yang terdiri dari 3 folder yaitu *bin* (exe), *src* (kode program), dan *doc* (laporan pdf dan *readme.txt*)

Tempat pengumpulan *hardcopy*: Di atas loker Lab IRK

Tempat pengumpulan *softcopy*: *Dropbox* (akan diumumkan lewat milis)

Deskripsi tugas :

Pada tugas kali 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**.

Berikutnya, program dapat menerima input yang didefinisikan pada tabel dibawah.

Input	Keterangan
<code>translate <dx> <dy></code>	Melakukan translasi objek dengan menggeser nilai x sebesar <i>dx</i> dan menggeser nilai y sebesar <i>dy</i> .
<code>dilate <k></code>	Melakukan dilatasi objek dengan faktor scaling <i>k</i> .

<code>rotate <deg> <a> </code>	Melakukan rotasi objek secara berlawanan arah jarum jam sebesar <i>deg</i> derajat terhadap titik <i>a,b</i>
<code>reflect <param></code>	Melakukan pencerminan objek. Nilai <i>param</i> 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.
<code>shear <param> <k></code>	Melakukan operasi <i>shear</i> pada objek. Nilai <i>param</i> dapat berupa <i>x</i> (terhadap sumbu x) atau <i>y</i> (terhadap sumbu y). Nilai <i>k</i> adalah faktor <i>shear</i> .
<code>stretch <param> <k></code>	Melakukan operasi <i>stretch</i> pada objek. Nilai <i>param</i> dapat berupa <i>x</i> (terhadap sumbu x) atau <i>y</i> (terhadap sumbu y). Nilai <i>k</i> adalah faktor <i>stretch</i> .
<code>custom <a> <c> <d></code>	Melakukan transformasi linier pada objek dengan matriks transformasi sebagai berikut: $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
<code>multiple <n></code> <code>... // input 1</code> <code>... // input 2</code> <code>...</code> <code>... // input n</code>	Melakukan transformasi linier pada objek sebanyak <i>n</i> kali berurutan. Setiap baris input 1.. <i>n</i> dapat berupa <i>translate, rotate, shear, dll</i> tetapi bukan <i>multiple, reset, exit</i> .
<code>reset</code>	Mengembalikan objek pada kondisi awal objek didefinisikan.
<code>exit</code>	Keluar dari program.

Contoh I/O program :

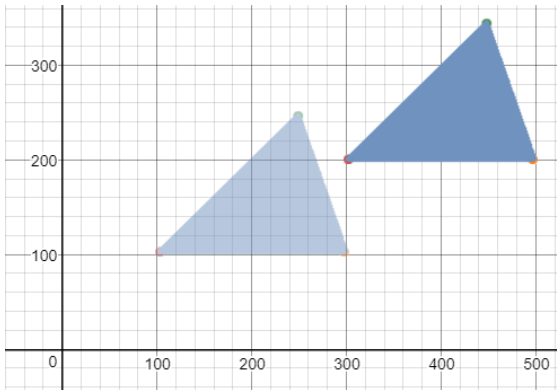
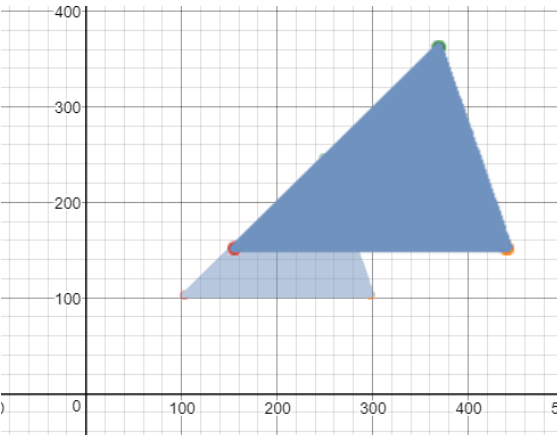
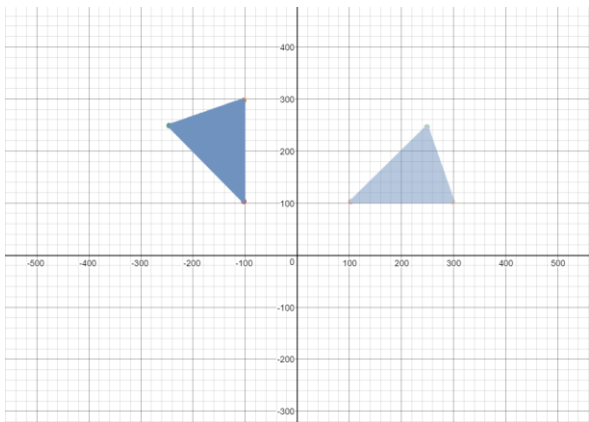
Saat program baru dimulai:

Input	Output
<pre>3 100,100 250,250 300,100</pre>	

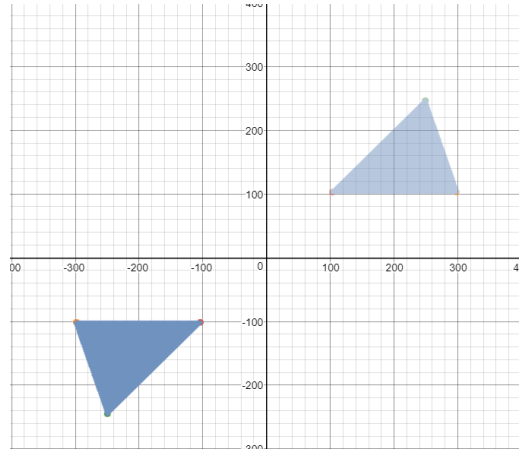
Perhatikan bahwa garis-garis tipis pada gambar diatas *tidak perlu diimplementasikan* pada program.

Saat program sudah membentuk objek dari input awal:

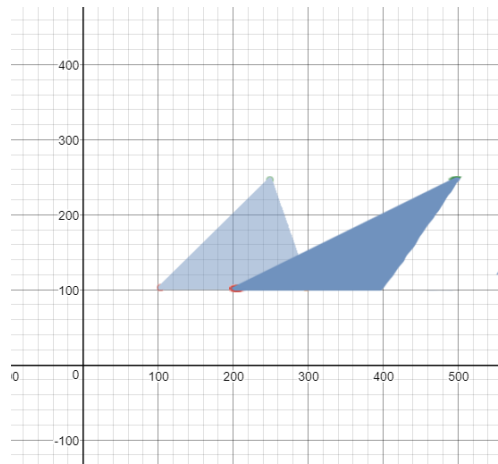
Catatan: Perhatikan bahwa gambar bidang yang transparan menunjukkan kondisi bidang *sebelum* input diberi, sedangkan bidang yang tidak transparan menunjukkan kondisi bidang *setelah* program mengeksekusi operasi dari input (bidang yang transparan *tidak ditampilkan* pada program).

Input	Output
<pre>translate 200 100</pre>	
<pre>dilate 1.5</pre>	
<pre>rotate 90 0 0</pre>	

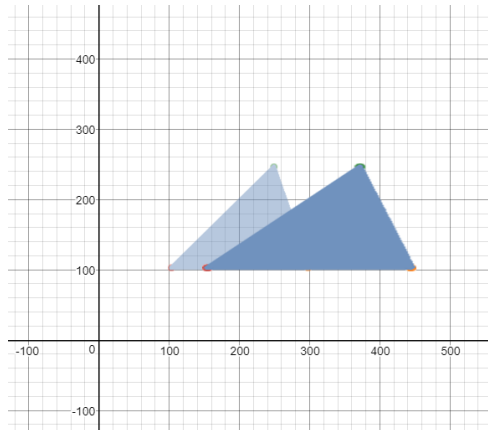
reflect (0,0)

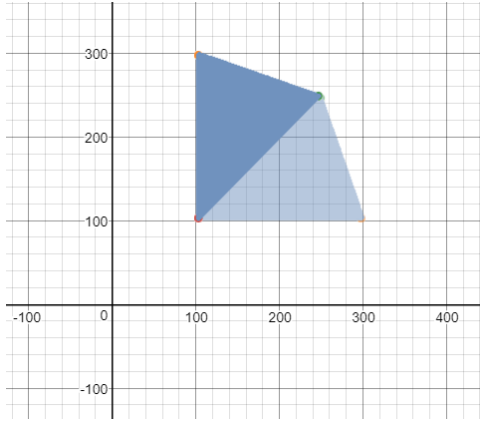
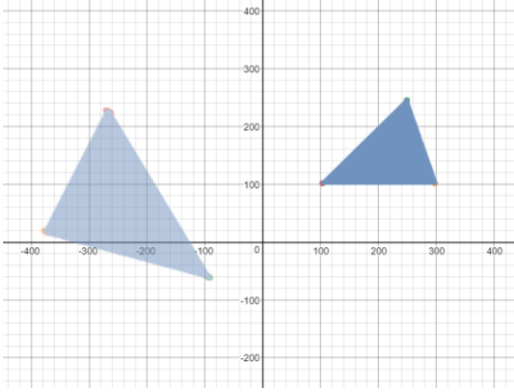


shear x 1



stretch x 1.5



<pre>custom 0 1 1 0</pre>	
<pre>reset</pre>	

Penjelasan singkat OpenGL API

Open Graphics Library (OpenGL) adalah API (*Application Programming Interface*) yang berfungsi untuk melakukan *rendering* grafik 2D dan 3D. *OpenGL* bersifat *cross-language*, *cross-platform*, dan *open source*. *OpenGL* umumnya digunakan untuk melakukan interaksi dengan GPU (*graphics processing unit*) untuk mencapai hasil *render* yang diakselerasi dengan *hardware*. Anda diharapkan untuk melakukan eksplorasi penggunaan *OpenGL*. Berikut adalah contoh kode program yang menggunakan library *OpenGL*:

Kode Program (khusus untuk objek 2D)	Keterangan
<pre>GLfloat triangleVertices[] = { 100, 100, 0, 300, 100, 0, 250, 250, 0 };</pre>	<p>Mendefinisikan tiga buah titik, yaitu (100,100,0), (300,100,0), dan (250,250,0). Perhatikan nilai ketiga dari titik adalah nol supaya titik berupa 2D.</p>
<pre>GLFWwindow *window; window = glfwCreateWindow (600, 600, "MyWindowName", NULL, NULL);</pre>	<p>Membuat sebuah window yang akan Anda gunakan untuk menampilkan output program.</p>
<pre>glEnableClientState(GL_VERTEX_ARRAY); glVertexPointer(3, GL_FLOAT, 0, triangleVertices); glDrawArrays(GL_POLYGON, 0, 3);</pre>	<p>Menggambar sebuah poligon sesuai titik-titik yang sudah didefinisikan pada <code>triangleVertices</code>.</p>

```
glDisableClientState(GL_VERTEX_ARRAY);
```

Berikut adalah daftar pranala yang dapat membantu Anda untuk melakukan eksplorasi OpenGL:

1. Tutorial OpenGL: <http://www.opengl-tutorial.org/>
2. Wiki: https://www.khronos.org/opengl/wiki/Getting_Started
3. Library: <https://www.opengl.org/sdk/libs/>
4. Wikipedia: <https://en.wikipedia.org/wiki/OpenGL>

SPESIFIKASI TUGAS

1. Program ini harus Anda buat dalam Bahasa *Python*, dan harus menggunakan OpenGL API.
2. Untuk melakukan transformasi linier (translasi, rotate, dsb), tidak boleh menggunakan fungsi-fungsi yang ada di dalam OpenGL (misalnya *glTranslate*, *glRotate*, dsb), tetapi menggunakan perkalian matriks transformasi dengan vector.
3. Selain untuk objek 2D, program harus dapat mensimulasikan transformasi linier tiga dimensi (input titik menjadi nilai **x**, **y**, dan **z**). Silahkan Anda definisikan sendiri bagaimana cara untuk menangani kasus 3D (misalnya operasi translasi perlu menerima parameter dz).
4. Kreasikan objek 2D dan 3D sesuka dan sekreatifmu. Koordinat objek 2D adalah input dari pengguna, sedangkan untuk objek 3D boleh *hard coded* di dalam program.
5. **Bonus:** Program dapat menampilkan animasi ketika melakukan operasi transformasi linier. Contohnya ketika program melakukan operasi rotasi, program menampilkan animasi gerakan rotasi objek dari posisi awal sampai posisi akhir.

LAPORAN

1. *Cover:* *Cover* laporan ada foto anggota kelompok (foto bertiga, bebas gaya). Foto ini menggantikan logo “gajah” ganesha.
Bab 1: Deskripsi masalah (dapat meng-*copy paste* file tugas ini)
Bab 2: Teori singkat mengenai transformasi linier dan matriks transformasi, OpenGL, dll.
Bab 3: Implementasi program dalam Bahasa pemrograman yang dipilih. Pembagian tugas antar anggota kelompok.
Bab 4: Eksperimen. Bab ini berisi hasil eksekusi program terhadap contoh-contoh transformasi obyek gambar, screen shot hasil eksekusi, dan analisis hasilnya
Bab 5: Kesimpulan (hasil yang dicapai), saran (saran pengembangan), refleksi (komentar anda terhadap tugas ini).

Tuliskan juga referensi (buku, web), yang dipakai/diacu di dalam Daftar Referensi.

Keterangan laporan dan program:

- a) Laporan ditulis dalam bahasa Indonesia yang baik dan benar, tidak perlu panjang tetapi tepat sasaran dan jelas.

- b) Laporan tidak perlu memakai *cover* mika dan dijilid. Cukup dibuat agar laporan tidak akan tercecer bila dibaca.
- c) Laporan boleh menggunakan kertas riu, boleh bolak-balik, boleh dalam satu halaman kertas terdapat dua halaman tulisan asalkan masih terbaca.
- d) Identitas per halaman harus jelas (misalnya : halaman, kode kuliah).
- e) *Listing* program ataupun algoritma tidak perlu disertakan pada laporan.
- e) Program disimpan di dalam *folder* Algeo-xxxxx. Lima digit terakhir adalah NIM anggota terkecil. Didalam *folder* tersebut terdapat tiga folder bin, src dan doc yang masing-masing berisi :

Folder *bin* berisi *executable file*

Folder *src* berisi *source code*

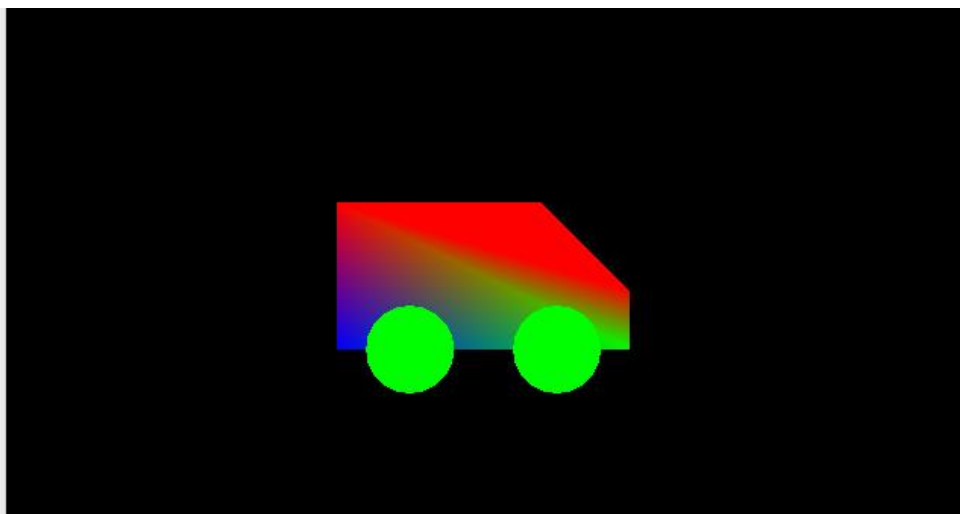
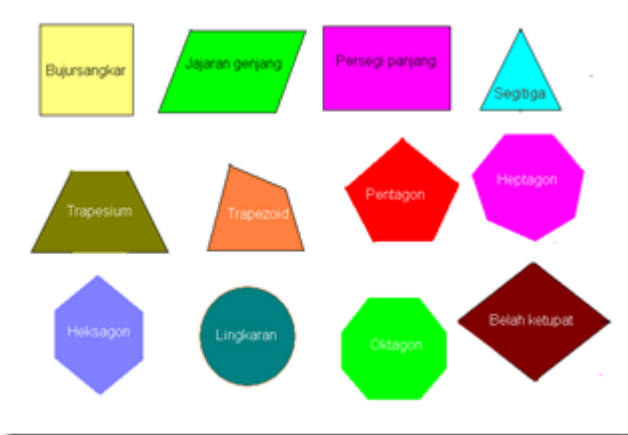
Folder *doc* berisi dokumentasi program dan *readme*

Lain – lain :

1. Tugas dikerjakan per kelompok dengan jumlah anggota 3 orang, tidak boleh sama dengan kelompok pada tugas besar sebelumnya, boleh lintas kelas.
2. Program harus modular dan mengandung komentar yang jelas.
3. Dilarang menggunakan kode program yang diunduh dari Internet. Mahasiswa harus membuat program sendiri, tetapi belajar dari program yang sudah ada tidak dilarang.
4. Pengumpulan laporan Tubes 2 adalah hari Jumat tanggal 19 November 2018 sebelum pukul 11.00. Keterlambatan akan mengurangi nilai.
5. *Source code* disimpan pada folder *src*, *file executable* disimpan pada folder *bin*, laporan pdf dan *readme.txt* disimpan pada folder *doc*. *Readme.txt* menjelaskan cara melakukan kompilasi dan eksekusi program secara singkat.
6. Semua pertanyaan menyangkut tugas ini harus dikomunikasikan melalui milis agar dapat dicermati oleh semua peserta kuliah IF2123.
7. *Soft-copy* (3 folder di atas) akan dikirim ke *dropbox* (akan diumumkan kemudian), tetapi *hardcopy* laporan (tanpa CD/*flashdisk*) tetap harus dikumpulkan di atas loker IRK.

Tambahan:

Contoh-contoh objek 2D:



Contoh-contoh objek 3D:



