

Nama : Augustav Fahrul Alzaiya
NIM : 2200018048
Prak : Grafkom 12:15

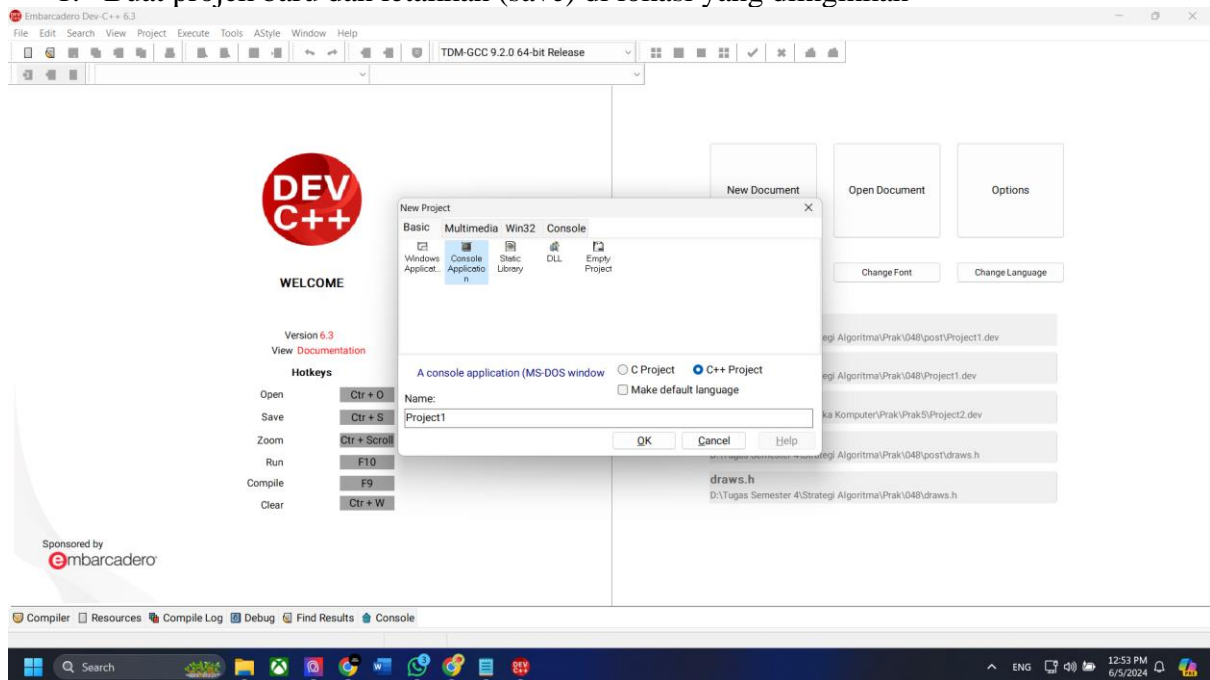
- Pretest 8

Nama: Augustav Fahrul A. NIM : 2200018048	Asisten: Paraf Asisten:	Tanggal: 5-6-2024 Nilai:
--	----------------------------	-----------------------------

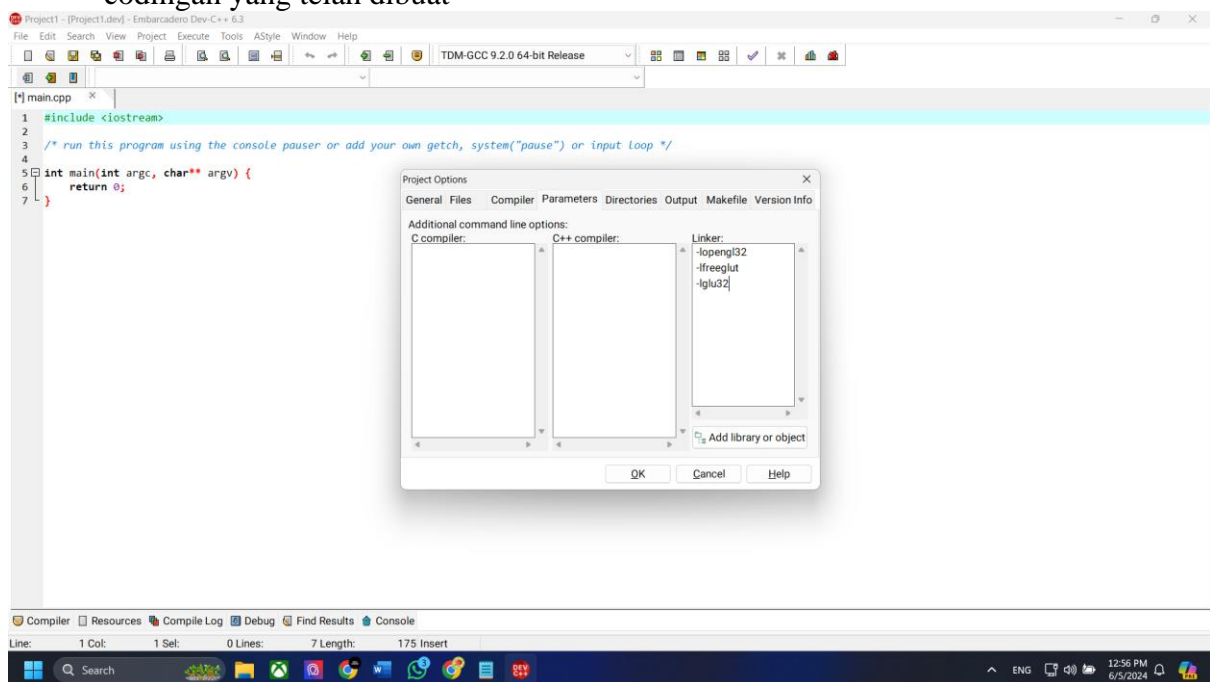
1. Boundary Representation adalah salah satu metode dalam representasi objek 3D yang menggambarkan bentuk dan struktur objek melalui batas-batas atau permukaannya.
2. a. Polygon mesh \rightarrow Permukaan objek dibentuk oleh sekumpulan polygon, biasanya segitiga atau segi empat yang saling berhubungan.
b. NURBS (Non-Uniform Rational B-Splines)
 \rightarrow menggunakan kurva dan permukaan yang dapat dikontrol dengan kontrol point, knot vector, dan weight.
c. Subdivision Surfaces \rightarrow Teknik yang memulai dengan sebuah mesh kasar dan kemudian secara rekursif membagi setiap polygon untuk menghasilkan permukaan yang lebih halus.
3. Kurva NURBS \rightarrow generalisasi dari B-spline yang menambahkan weight ke setiap kontrol point, memungkinkan kontrol yang lebih besar, dan fleksibilitas dalam menggambarkan bentuk kompleks.
 - NURBS \rightarrow menggunakan weight, lebih fleksibel, lebih kompleks
 - B-Splines \rightarrow tidak menggunakan weight, kurang fleksibel dan lebih sederhana.

- Langkah Praktikum 8

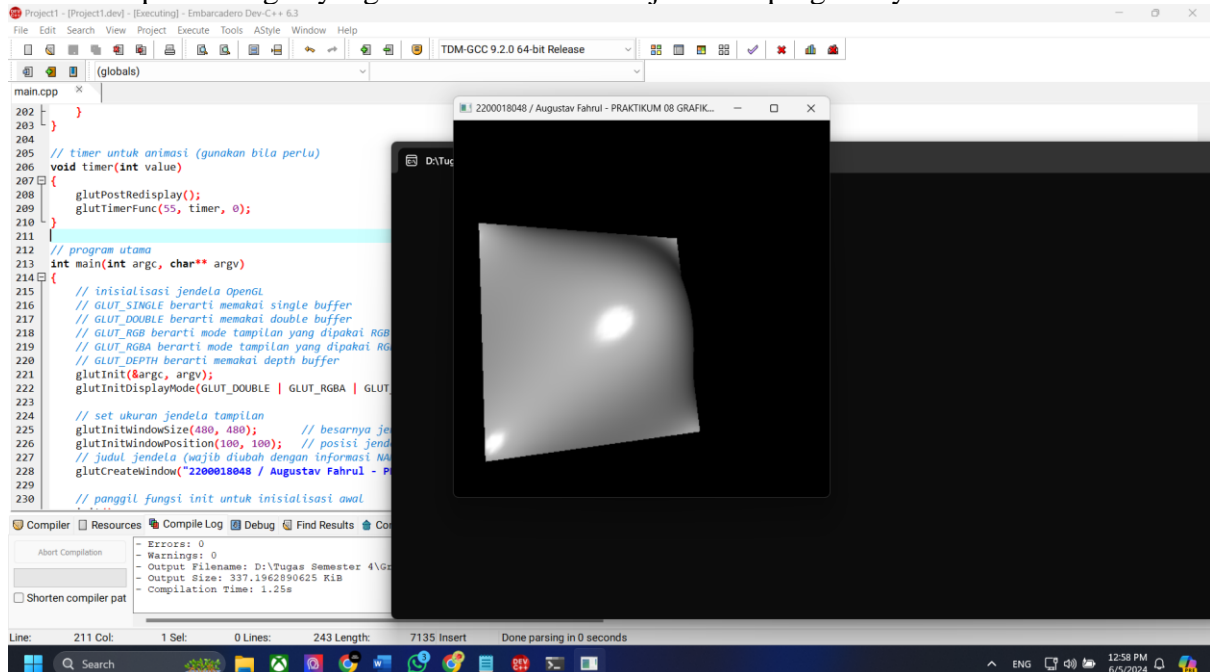
1. Buat proyek baru dan letakkan (save) di lokasi yang diinginkan



2. Kemudian masukkan linker. Linker ini berguna untuk membaca gambar gambar dari codingan yang telah dibuat

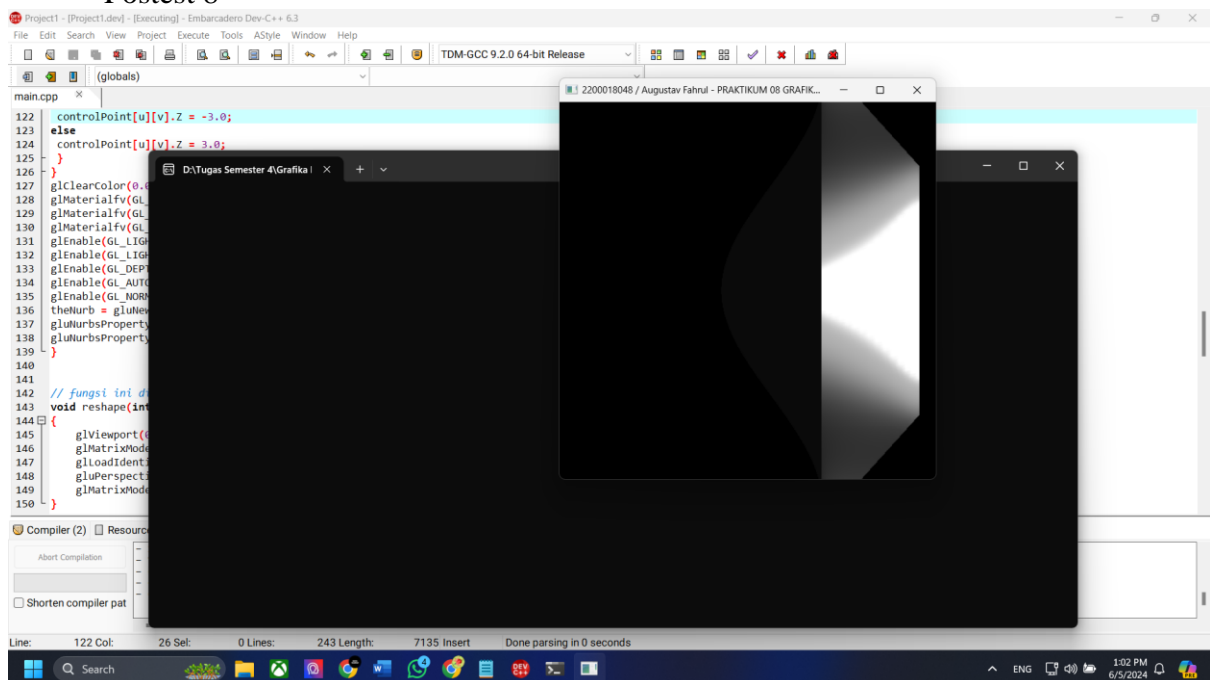


3. Copas codingan yang diberikan kemudian jalankan programnya



Pada codingan diatas, asisten memberikan codingan yang dimana membuat sebuah gambar yang mencekung ke depan. Poin paling penting pada cekungan ini ada pada bagian void init(void), karena pada bagian tersebut ada codingan yang mendeklarasikan gambarnya mencekung ke dalam atau keluar, rata atau tidaknya gambar tersebut.

- Postest 8



Pada bagian posttest, asisten meminta gambarnya agar menjadi rata, caranya ialah dengan mengganti sedikit angka yang ada di codingan berikut :

```
if ((u == 1 || u == 2) && (v == 1 || v == 2))  
    controlPoint[u][v].Z = 3.0;  
else  
    controlPoint[u][v].Z = -3.0;  
}  
}
```

Menjadi

```
if ((u == 1 || u == 2) && (v == 1 || v == 2))  
    controlPoint[u][v].Z = -3.0;  
else  
    controlPoint[u][v].Z = 3.0;  
}  
}
```

Jadi pada codingan diatas, diinisialisasikan untuk yang awal yaitu minusnya di point kedua(dalam else) maka gambar akan mencekung ke depan atau pada bagian yang terang.

Dan jika codingan diatas diganti minusnya ke bagian if dan else nya menjadi positif, maka gambar akan mencekung ke belakang seperti gambar yang sudah kita lihat.