

# **LAPORAN PRATIKUM GRAFIK KOMPUTER**

Diajukan untuk memenuhi Tugas mata kuliah Pratikum Grafik Komputer

## **IMPLEMENTASI OPENGL DALAM MEMBUAT OBJEK 3D RUANGAN KELAS**

Dosen Pengampu : Sri Rahayu, M.Kom

Instruktur Pratikum : Arul Budi Kalimat, S.Kom



Disusun oleh

Kelompok : 5

Alamahul Bayan

2306163

Agna Fadia

2306145

Ramdhani Sulaeman

Burhanudin

2306163

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN ILMU KOMPUTER**

**INSTITUT TEKNOLOGI GARUT**

**2025**

## **KATA PENGANTAR**

Puji dan syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya sehingga kami dapat menyelesaikan Laporan Praktikum Jaringan Komputer ini. Laporan ini dibuat sebagai salah satu tugas dari mata kuliah Jaringan Komputer, dengan tujuan untuk memberikan pemahaman yang lebih baik tentang Implementasi OpenGL dalam Membuat Objek 3D Ruang Kelas.

Kami mengucapkan terima kasih kepada dosen pengampu Sri Rahayu, M.Kom, instruktur praktikum Arul Budi Kalimat, S.Kom, serta semua pihak yang telah memberikan dukungan dalam penyusunan laporan ini.

Kami menyadari bahwa laporan ini masih memiliki kekurangan, untuk itu kami mengharapkan kritik dan saran yang membangun demi perbaikan di masa yang akan datang.

Garut, 4 Januari 2025

Kelompok 5

## DAFTAR ISI

KATA PENGANTAR .....	i
DAFTAR ISI .....	2
DAFTAR GAMBAR.....	3
BAB I PENDAHULUAN .....	4
1.1 Latar Belakang .....	4
1.2 Rumusan Masalah .....	5
1.3 Tujuan .....	5
BAB II TINJAUAN PUSTAKA.....	3
2.1 OpenGL .....	3
2.2 Konfigurasi OpenGL pada Dev C++ .....	3
2.3 Membuat Objek 3D Ruang Kelas di OpenGL .....	10
BAB III HASIL .....	12
3.1 Source Code .....	12
3.2 Output.....	12
3.3 Penjelasan .....	13
BAB IV KESIMPULAN .....	16
DAFTAR PUSTAKA.....	17

## DAFTAR GAMBAR

Gambar 1 Tampilan Dev C++ yang terinstall.....	4
Gambar 2 Hasil Download File Praktikum.....	4
Gambar 3 Tampilan Archive File Praktikum 1.....	5
Gambar 4 Tampilan setelah masuk ke folder Freeglut .....	5
Gambar 5 Hasil ekstrak folder lib .....	6
Gambar 6 Hasil Ekstrak folder include .....	6
Gambar 7 Hasil ekstrak file freeglut .....	7
Gambar 8 Tampilan membuat Project di Dev c++ .....	8
Gambar 9 Menentukan jenis project di Dev C++ .....	8
Gambar 10 Mengkonfigurasi project di Dev C++ .....	9
Gambar 11 Menambahkan linker di Dev C++.....	9
Gambar 12 Telah selesai konfigurasi OpenGL di Dev C++.....	10
Gambar 13 Output kodingan didalam dengan lampu mati .....	12
Gambar 14 Output kodingan didalam dengan lampu hidup .....	13
Gambar 15 Output kodingan diluar dengan lampu mati.....	13

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pembuatan objek tiga dimensi (3D) merupakan salah satu aspek penting dalam pengembangan aplikasi grafis, animasi, dan simulasi. Dalam dunia informatika, kemampuan untuk memodelkan dan memvisualisasikan objek 3D digunakan secara luas di berbagai bidang, seperti desain arsitektur, pengembangan permainan, simulasi fisika, dan pembelajaran interaktif. Teknologi seperti OpenGL (Open Graphics Library) memungkinkan pengembang untuk membuat visualisasi grafis yang realistis dengan efisiensi tinggi[1].

Praktikum ini bertujuan untuk memberikan pemahaman dasar kepada mahasiswa mengenai konsep dan implementasi teknologi grafis 3D menggunakan OpenGL. Dengan memahami cara kerja OpenGL, mahasiswa dapat mengaplikasikan konsep-konsep seperti rendering, pencahayaan, dan transformasi geometris dalam pembuatan objek 3D. Salah satu aplikasi praktis dari keterampilan ini adalah pembuatan model ruangan kelas 3D[2].

Pentingnya penguasaan OpenGL juga didukung oleh banyak literatur dan penelitian. Menurut D. Shreiner et al., dalam buku "OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3", OpenGL adalah standar industri yang mendukung pengembangan aplikasi grafis lintas platform, sehingga memungkinkan implementasi solusi grafis yang fleksibel dan scalable [3]. Di sisi lain, J. Foley et al., dalam buku "Computer Graphics: Principles and Practice", menyebutkan bahwa pemodelan 3D dan simulasi grafis merupakan bagian penting dari pendidikan interaktif berbasis teknologi[3].

Selain itu, Hearn dan Baker dalam buku mereka "Computer Graphics with OpenGL" mengemukakan bahwa teknologi grafis 3D, termasuk OpenGL, memainkan peran kunci dalam menciptakan visualisasi interaktif yang memberikan pengalaman pengguna yang lebih imersif. Buku tersebut juga menekankan pentingnya pemahaman dasar terhadap pipeline grafis untuk menciptakan objek yang efisien dan estetis[2].

Melalui praktikum ini, mahasiswa diharapkan mampu memahami implementasi OpenGL dalam menciptakan objek 3D yang akurat dan estetis. Selain itu, pengalaman praktis ini juga dapat menjadi dasar untuk pengembangan keterampilan lebih lanjut dalam bidang grafis komputer dan teknologi virtual reality[4][2].

## **1.2 Rumusan Masalah**

1. Apa yang dimaksud OpenGL?
2. Bagaimana cara mengkonfigurasi OpenGL pada Dev C++?
3. Bagaimana cara kerja dari OpenGL?
4. Bagaimana membuat objek 3D Ruang Kelas dalam OpenGL?

## **1.3 Tujuan**

1. Mengetahui apa itu OpenGL.
2. Mengetahui cara mengkonfigurasi OpenGL pada Dev C++.
3. Mengetahui cara kerja dari OpenGL.
4. Mengetahui cara pembuatan objek 3D Ruang Kelas dalam OpenGL.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 OpenGL**

OpenGL (Open Graphics Library) adalah sebuah API (Application Programming Interface) yang digunakan untuk membuat aplikasi grafis dua dimensi (2D) dan tiga dimensi (3D) secara real-time. OpenGL berfungsi sebagai standar industri yang mendukung pengembangan aplikasi grafis lintas platform, sehingga mempermudah pengembang dalam menciptakan visualisasi yang efisien dan interaktif. Teknologi ini banyak digunakan dalam berbagai bidang, termasuk simulasi, animasi, desain arsitektur, dan pengembangan permainan[1].

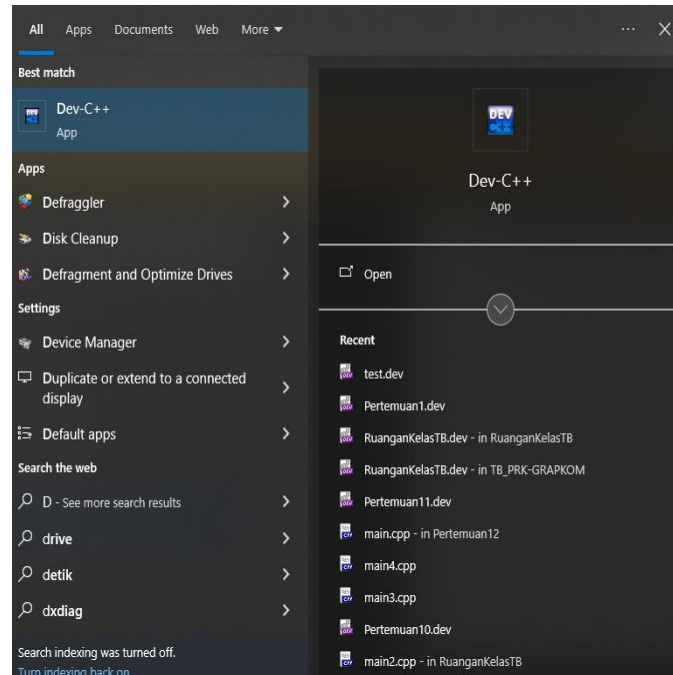
OpenGL dirancang untuk memberikan akses langsung ke perangkat keras grafis melalui serangkaian fungsi yang memungkinkan manipulasi objek, pengaturan pencahayaan, tekstur, dan rendering. Salah satu keunggulan utama OpenGL adalah sifatnya yang platform-independent, sehingga aplikasi yang dikembangkan dapat dijalankan di berbagai sistem operasi tanpa modifikasi besar[2].

Menurut Shreiner et al., OpenGL menggunakan pipeline grafis sebagai model pemrosesan data untuk menghasilkan gambar dari objek 3D. Pipeline ini terdiri dari beberapa tahapan, seperti transformasi geometri, rasterisasi, dan fragment shading, yang bekerja bersama untuk menghasilkan visualisasi yang realistis[4].

#### **2.2 Konfigurasi OpenGL pada Dev C++**

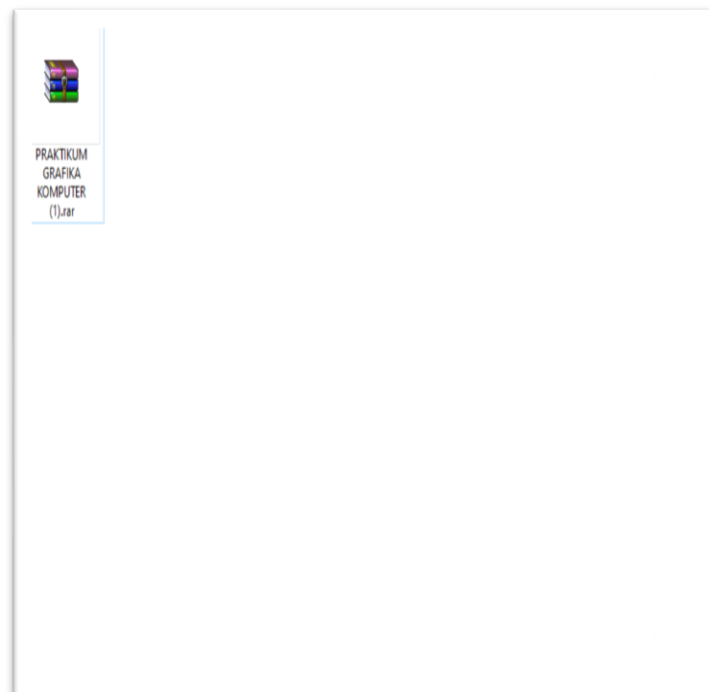
Uraikan dan Jelaskan langkah-langkah untuk konfigurasi OpenGL pada Dev C++. Setiap langkah harus menyertakan Screenshoot. Semua gambar ukurannya harus sama dan posisi gambar berada ditengah (center) dan berikan keterangan dari setiap gambar. Contoh :

1. Pastikan Dev C++ terinstall.



**Gambar 1 Tampilan Dev C++ yang terinstall**

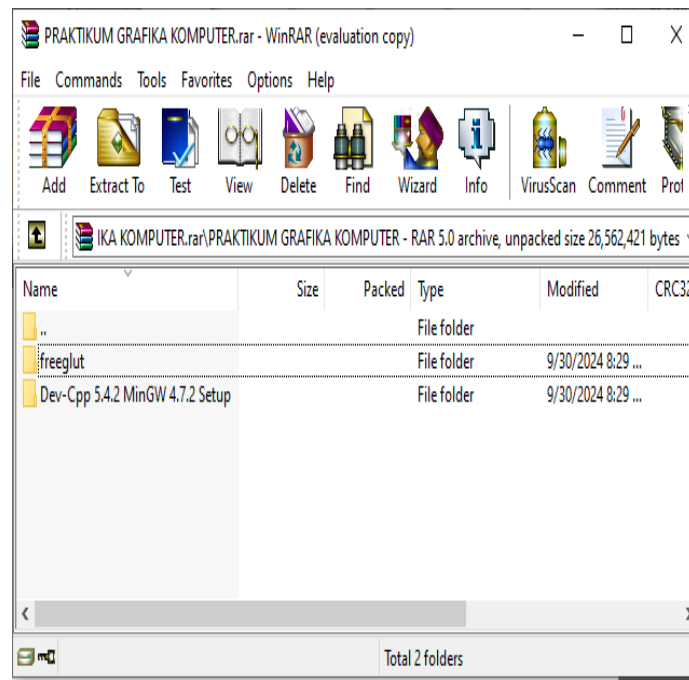
2. Lalu download file archive praktikum yang telah disediakan oleh praktikum.



**Gambar 2 Hasil Download File Praktikum**

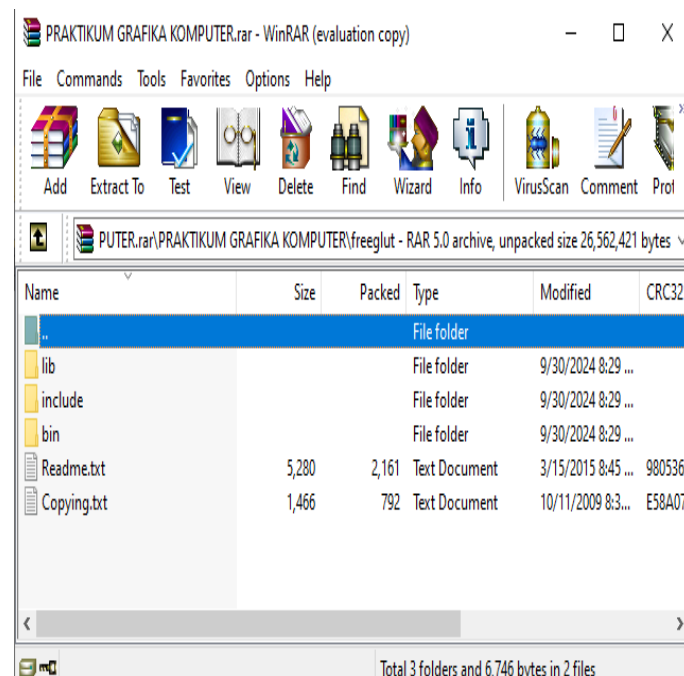


3. Lalu masuk ke dalam file rar.



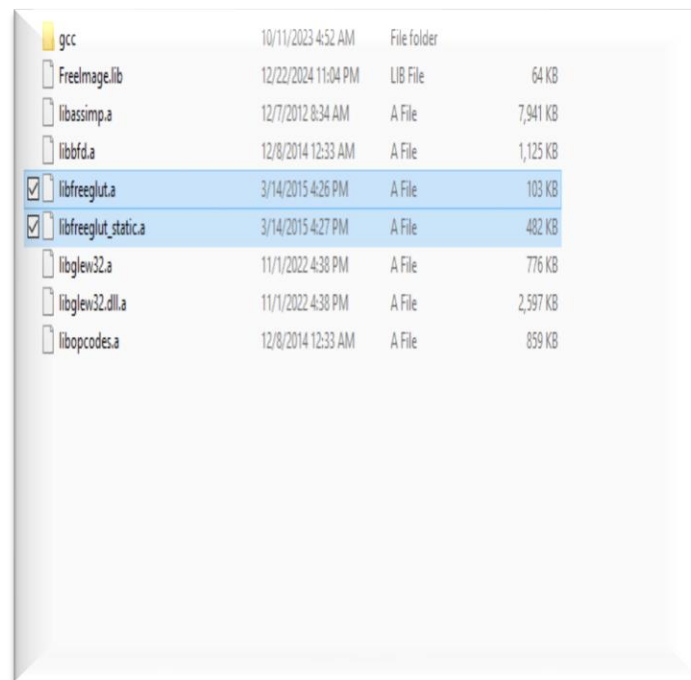
**Gambar 3 Tampilan Archive File Praktikum 1**

4. Lalu masuk ke folder freelut.



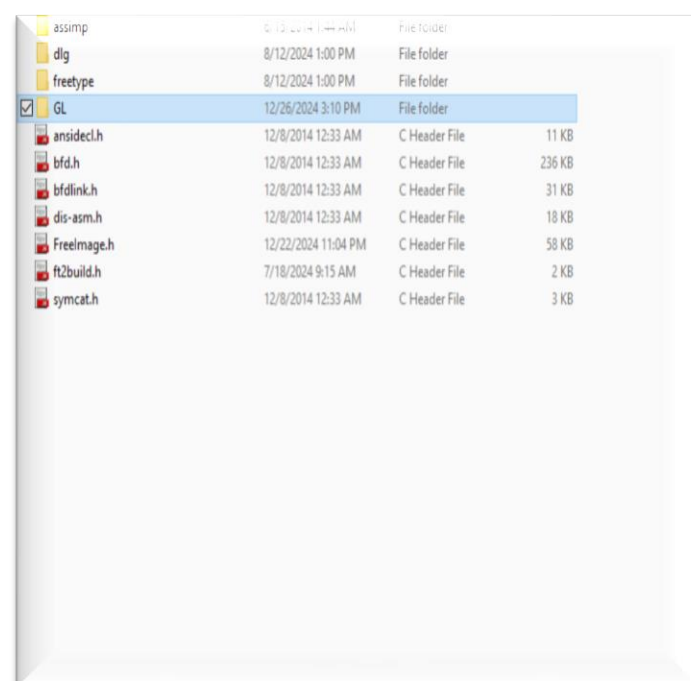
**Gambar 4 Tampilan setelah masuk ke folder Freelut**

5. Extrak folder lib ke “[tempat instalasi Dev C++]/MinGW/” jika perangkat computernya 64bit maka Extrak folder lib/64 ke “(tempat instalasi Dev C++)/MinGW/lib”.



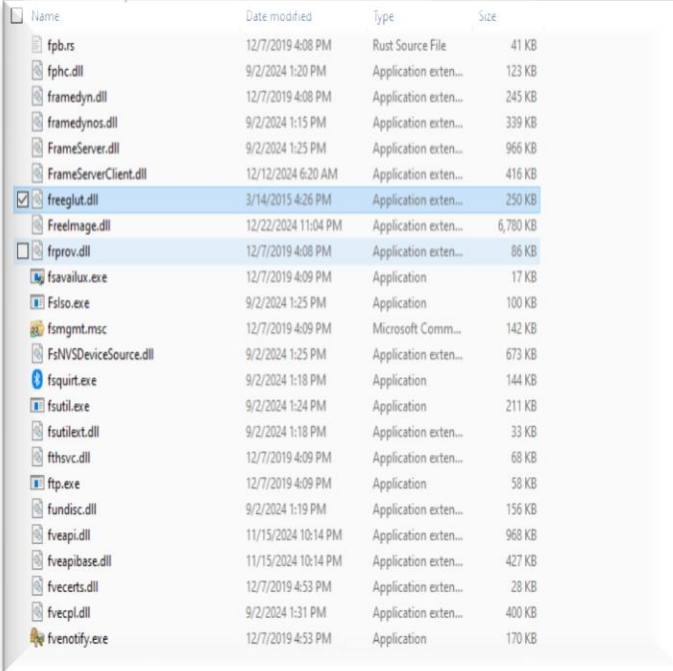
**Gambar 5 Hasil ekstrak folder lib**

6. Masih di File Rar, ekstrak folder include “[tempat instalasi Dev C++]/MinGW/”.



**Gambar 6 Hasil Extrak folder include**

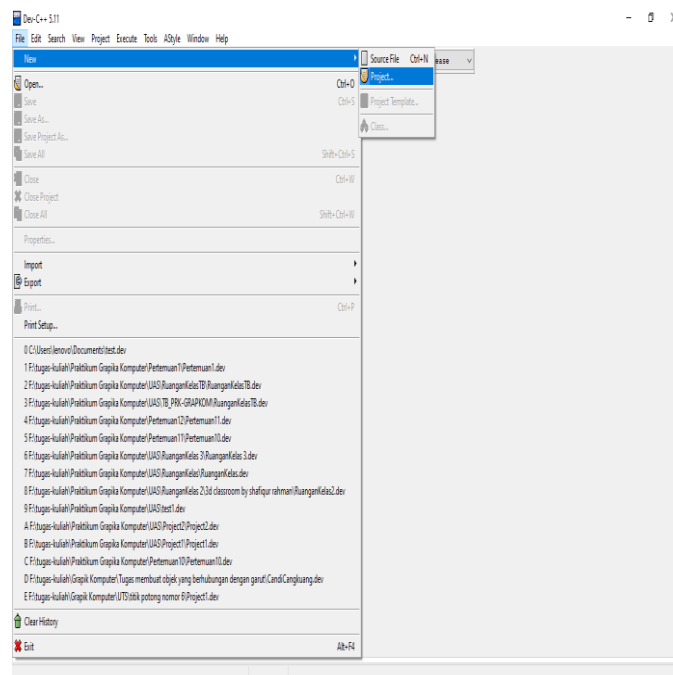
7. Masih di file Rar, ekstrak folder bin ke “C://Windows//System32/” jika perangkat computernya 64bit maka Extrak folder “bin/64” ke “C://Windows//System32/”.



Name	Date modified	Type	Size
fpb.rs	12/7/2019 4:08 PM	Rust Source File	41 KB
fpbc.dll	9/2/2024 1:20 PM	Application exten...	123 KB
framedyn.dll	12/7/2019 4:08 PM	Application exten...	245 KB
framedynos.dll	9/2/2024 1:15 PM	Application exten...	339 KB
FrameServer.dll	9/2/2024 1:25 PM	Application exten...	966 KB
FrameServerClient.dll	12/12/2024 6:20 AM	Application exten...	416 KB
<input checked="" type="checkbox"/> freelut.dll	3/14/2015 4:26 PM	Application exten...	250 KB
FreeImage.dll	12/22/2024 11:04 PM	Application exten...	6,780 KB
<input type="checkbox"/> frprov.dll	12/7/2019 4:08 PM	Application exten...	86 KB
fsavailux.exe	12/7/2019 4:09 PM	Application	17 KB
FsIso.exe	9/2/2024 1:25 PM	Application	100 KB
fsmgmt.msc	12/7/2019 4:09 PM	Microsoft Comm...	142 KB
FsNVSDeviceSource.dll	9/2/2024 1:25 PM	Application exten...	673 KB
fsquit.exe	9/2/2024 1:18 PM	Application	144 KB
fsutil.exe	9/2/2024 1:24 PM	Application	211 KB
fsutiltext.dll	9/2/2024 1:18 PM	Application exten...	33 KB
fthsvc.dll	12/7/2019 4:09 PM	Application exten...	68 KB
ftp.exe	12/7/2019 4:09 PM	Application	58 KB
fundisc.dll	9/2/2024 1:19 PM	Application exten...	156 KB
fveapi.dll	11/15/2024 10:14 PM	Application exten...	968 KB
fveapibase.dll	11/15/2024 10:14 PM	Application exten...	427 KB
fvecerts.dll	12/7/2019 4:53 PM	Application exten...	28 KB
fvecpl.dll	9/2/2024 1:31 PM	Application exten...	400 KB
fvenotify.exe	12/7/2019 4:53 PM	Application	170 KB

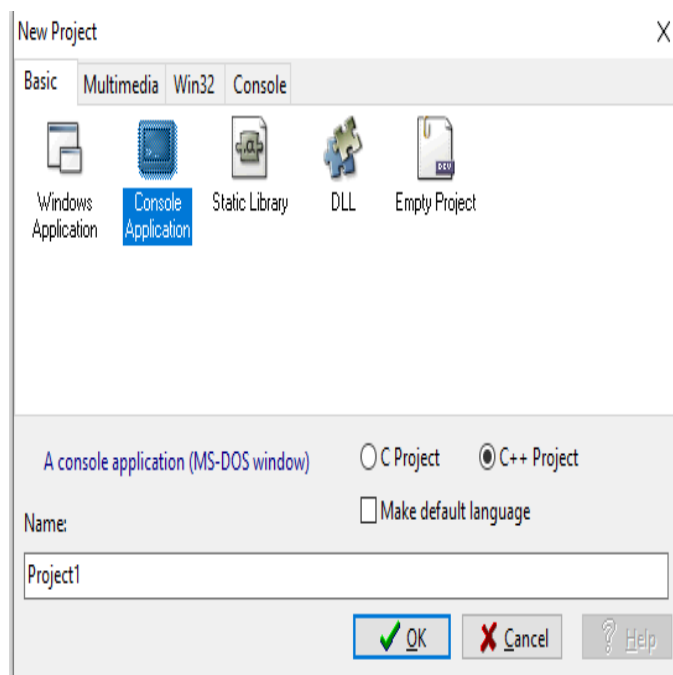
**Gambar 7 Hasil ekstrak file freelut**

8. Setelah proses ekstrak selesai masuk Dev C++ Lalu buat Project.



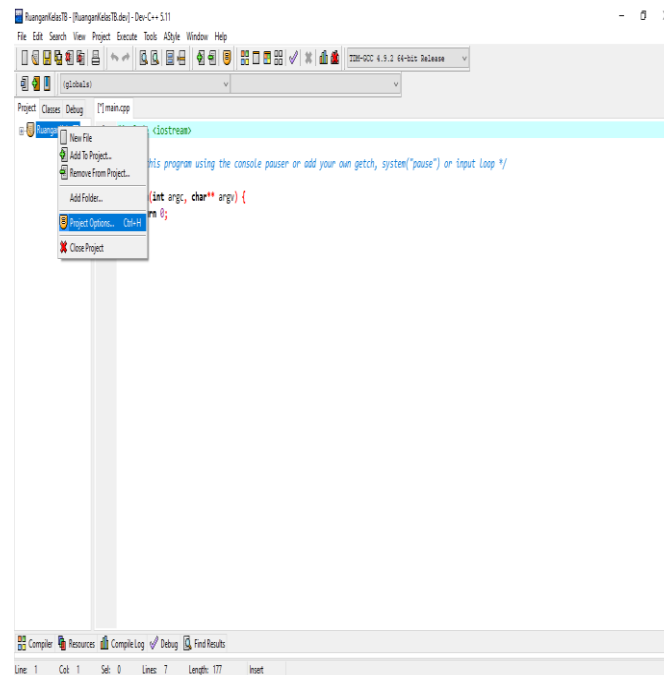
**Gambar 8 Tampilan membuat Project di Dev c++**

9. Di tab “basic” pilih console Application lalu tentukan nama projectnya.



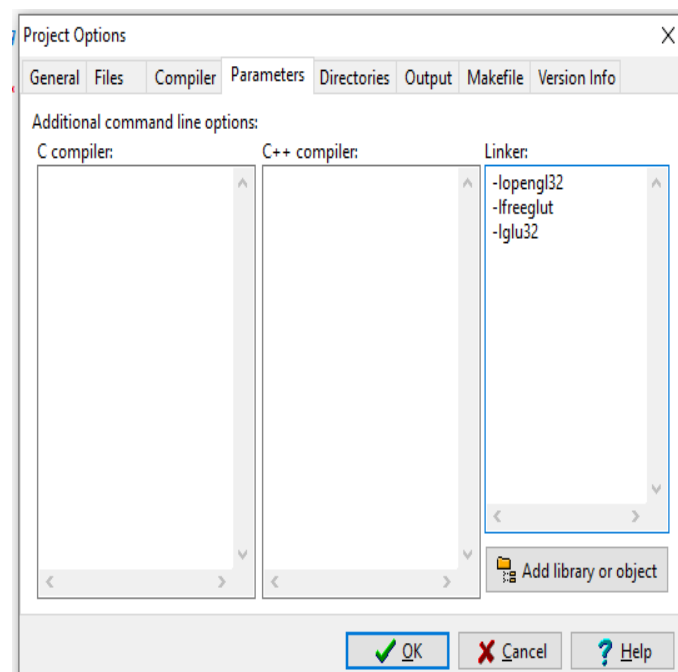
**Gambar 9 Menentukan jenis project di Dev C++**

10. Setelah itu klik kanan di projectnya lalu pilih “Project Option”.



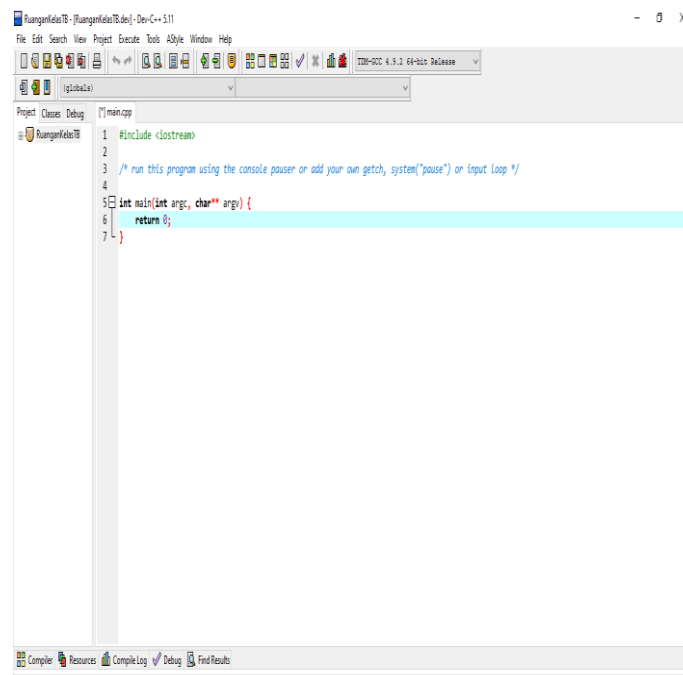
**Gambar 10 Mengkonfigurasi project di Dev C++**

11. Masuk ke tab “Parameters” dibagian linker isi parameter seperti gambar dibawah setelah itu klik “OK”.



**Gambar 11 Menambahkan linker di Dev C++**

12. Maka Konfigurasi Dev C++ telah selesai.



**Gambar 12** Telah selesai konfigurasi OpenGL di Dev C++

## 2.3 Membuat Objek 3D Ruang Kelas di OpenGL

Dalam pembuatan objek 3D ruangan kelas menggunakan OpenGL, dilakukan pemodelan berbagai elemen yang merepresentasikan ruangan kelas, seperti lantai, dinding, pintu, jendela, dan komponen pendukung lainnya. Berikut adalah konsep yang digunakan dalam proses pembuatan:

### 1. Pemodelan Objek dengan Fungsi Balok

- Fungsi `balok()` digunakan untuk menggambar objek berbentuk kubus dengan warna, kilauan, dan dimensi yang dapat disesuaikan. Objek ini menjadi dasar dalam membentuk elemen-elemen ruangan seperti dinding, lantai, atap, pintu, dan sebagainya.
- Material objek diatur melalui parameter `glMaterialfv()` untuk memberikan efek pencahayaan realistis, termasuk warna ambient, diffuse, dan specular.

### 2. Sistem Pencahayaan

- Fungsi `light()` mengatur pencahayaan dalam ruangan menggunakan sumber cahaya `GL_LIGHT0`. Posisi, warna ambient, diffuse, dan specular disesuaikan untuk menciptakan suasana yang lebih realistis.

- b. Pengguna dapat mengaktifkan atau menonaktifkan lampu menggunakan tombol tertentu.

### 3. Koordinat Kartesius

- a. Fungsi `drawCartecius()` digunakan untuk menampilkan koordinat kartesius sebagai referensi visual dalam pengaturan posisi objek. Sumbu X, Y, dan Z digambarkan menggunakan garis dan teks sebagai penanda.

### 4. Elemen Ruangan

- a. Lantai: Dibentuk dengan fungsi `lantai()`, memanfaatkan objek balok dengan dimensi dan posisi yang sesuai.
- b. Dinding: Fungsi `dinding_kiri()`, `dinding_kanan()`, `dinding_belakang()`, dan `dinding_depan()` digunakan untuk membangun struktur dinding ruangan.
- c. Atap: Dibuat dengan fungsi `atap()`, memberikan penutup untuk ruangan.
- d. Pintu: Fungsi `pintu()` membuat pintu dengan detail seperti daun pintu dan gagang.

### 5. Komponen Pendukung

- a. AC: Dibentuk menggunakan fungsi `AC()`, memanfaatkan balok dengan dimensi dan posisi yang menyerupai unit AC.
- b. Bor: Fungsi `bor()` merepresentasikan papan tulis di ruangan kelas.

### 6. Interaksi Pengguna

- a. Pengguna dapat mengatur posisi kamera menggunakan tombol navigasi (w, a, s, d, z, x) untuk melihat ruangan dari berbagai sudut.
- b. Rotasi ruangan diatur dengan tombol q dan e.
- c. Tombol k digunakan untuk menampilkan koordinat kartesius, sedangkan tombol l untuk mengontrol pencahayaan.

### 7. Main Loop

- a. Fungsi `main()` menginisialisasi jendela OpenGL dengan pengaturan awal seperti mode tampilan, ukuran jendela, dan callback untuk keyboard serta display.

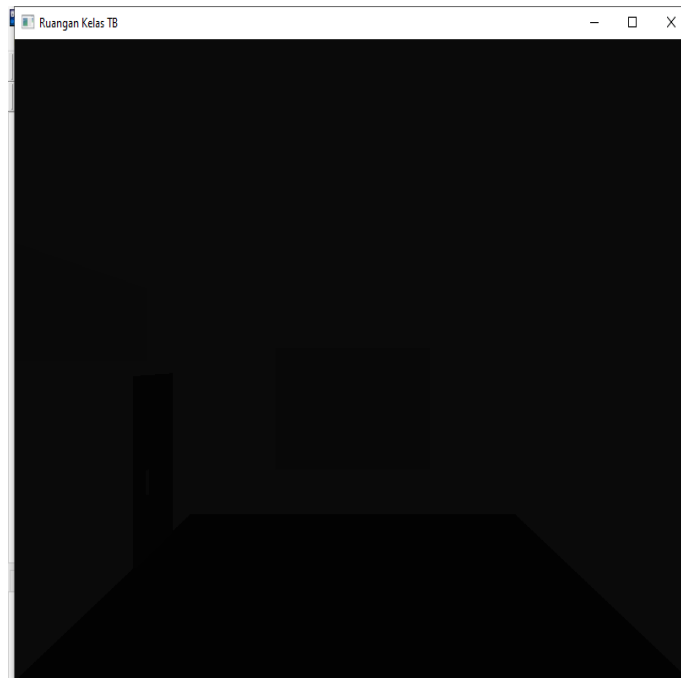
## BAB III

### HASIL

#### 3.1 Source Code

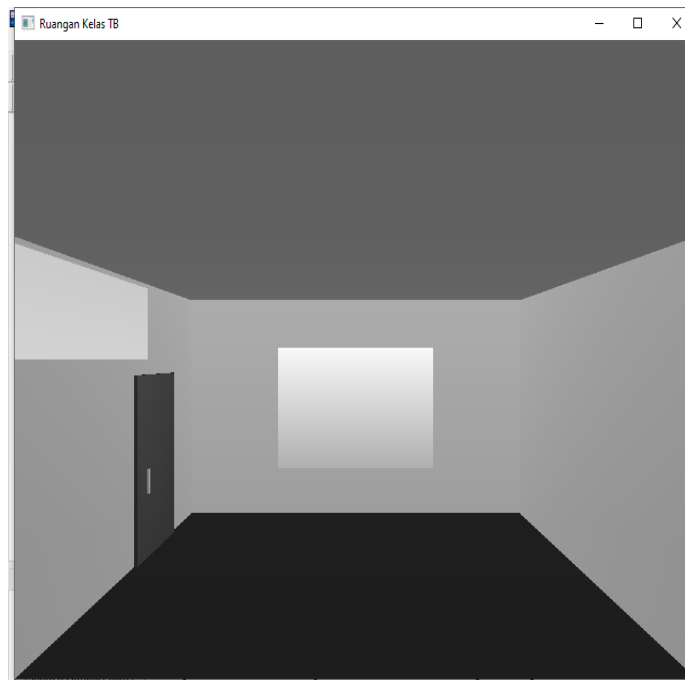
[https://github.com/Alamahul/TB\\_PRK-GRAPKOM](https://github.com/Alamahul/TB_PRK-GRAPKOM)

#### 3.2 Output

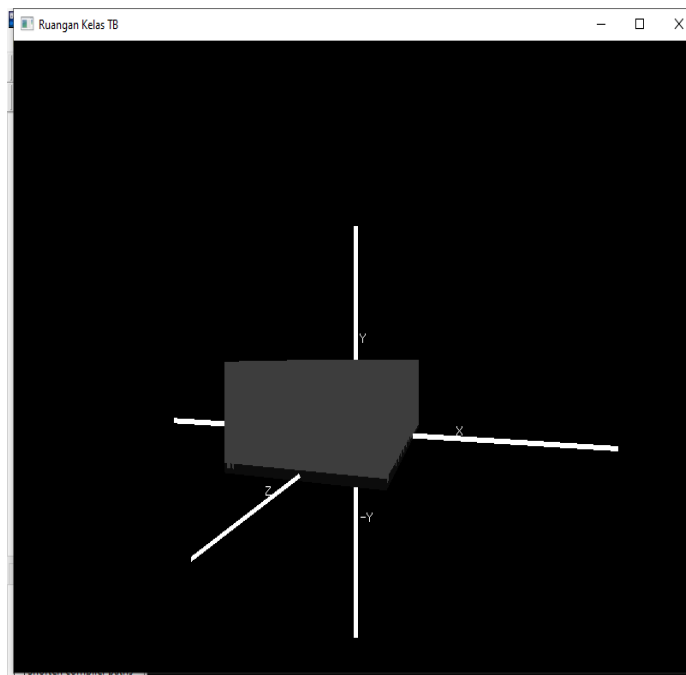


**Gambar 13** Output kodingan didalam ruangan dengan lampu mati





**Gambar 14 Output kodingan didalam ruangan dengan lampu hidup**



**Gambar 15 Output kodingan diluar ruangan dengan lampu mati**

### 3.3 Penjelasan

Source code ini merupakan implementasi grafis 3D menggunakan OpenGL dengan GLUT, yang menciptakan simulasi sebuah ruangan kelas. Berikut garis besar pembagian tugas dan penjelasan utama:

1. Pembagian Tugas
  - a. Ramdhani Sulaeman Burhanudin (2306161):  
Mengimplementasikan komponen utama struktur ruangan seperti lantai, dinding, atap, dan logika dasar program (fungsi utama dan loop utama).
  - b. Alamahul Bayan (2306163):  
Membuat fungsi tambahan untuk menggambar objek seperti pintu, sumbu koordinat (sistem Kartesius), dan pencahayaan. Selain itu, menyertakan fungsi untuk menampilkan teks di layar.
  - c. Agna Fadia (2306145):  
Menambahkan elemen dekorasi seperti AC dan mengintegrasikan elemen tersebut ke dalam ruangan dan juga mengatur kamera dan sudut pandang.
2. Struktur Utama Source Code
  - a. Header File dan Konstanta
    - 1) Library utama:  
`#include <GL/glut.h>` digunakan untuk membuat window dan menggambar elemen grafis.
    - 2) Konstanta dimensi:  
Lebar (`width`) dan tinggi (`height`) ruangan ditetapkan.
  - b. Variabel Global
    - 1) Variabel seperti `eyeX`, `eyeY`, `eyeZ` digunakan untuk mengontrol posisi kamera
    - 2) Boolean `lampu` dan `varKartesius` untuk mengaktifkan/mematikan pencahayaan dan menampilkan sumbu koordinat.
  - c. Fungsi Rendering
    - 1) `balok()`: Menggambar kubus sebagai dasar pembuatan objek seperti dinding, atap, dan pintu.
    - 2) `light()`: Mengatur pencahayaan ruangan dengan lampu yang dapat dinyalakan/dimatikan.
    - 3) `drawCartecius()`: Menampilkan sistem koordinat (sumbu X, Y, Z).
  - d. Komponen Ruangan
    - 1) Lantai dan Atap: Dibuat menggunakan fungsi `balok()` dengan dimensi spesifik.
    - 2) Dinding Kiri, Kanan, Depan, Belakang: Dibuat dengan transformasi geometris menggunakan `glTranslatef` dan `glScalef`.
    - 3) Pintu: Dibuat sebagai objek balok kecil dengan tambahan detail untuk pegangan.

e. Fungsi Event Handling

1) Fungsi `myKeyboardFunc()` menangani input dari keyboard untuk:

- a) Memutar kamera (dengan tombol `q` dan `e`).
- b) Menggeser kamera ke arah sumbu X, Y, dan Z.
- c) Mengaktifkan/mematikan lampu (`l`) dan sumbu Kartesius (`k`).

f. Fungsi Utama (`main()`)

1) Menginisialisasi GLUT, menciptakan window, mengatur event handler, dan memulai rendering loop.

## **BAB IV**

### **KESIMPULAN**

Dalam tugas besar ini, mahasiswa berhasil mempelajari penggunaan OpenGL untuk pembuatan objek 3D, khususnya dalam pemodelan ruangan kelas. Dengan memanfaatkan fungsi-fungsi dasar OpenGL, seperti pembuatan balok untuk menggambarkan elemen-elemen ruangan dan transformasi geometris untuk penataan posisi objek, mahasiswa dapat membuat representasi sederhana namun realistis dari sebuah ruangan. Konfigurasi lingkungan pengembangan menggunakan Dev C++ juga berjalan lancar, memungkinkan mahasiswa untuk memahami langkah-langkah teknis dalam mempersiapkan proyek OpenGL dengan benar.

Namun, tugas besar ini juga mengungkapkan tantangan yang ada dalam penggunaan OpenGL, seperti kompleksitas kode yang harus ditulis dan pemahaman yang lebih mendalam tentang konsep-konsep grafis, seperti pencahayaan, koordinat kartesius, dan transformasi objek. Meskipun demikian, keunggulan OpenGL dalam memberikan fleksibilitas dan kemampuan grafis yang tinggi sangat jelas. Ke depannya, disarankan untuk terus berlatih dengan proyek-proyek grafis lainnya untuk memperdalam pemahaman tentang teknik-teknik lanjutan, seperti shading dan teksturing, serta mengeksplorasi penggunaan IDE yang lebih modern untuk meningkatkan produktivitas pengembangan

## DAFTAR PUSTAKA

- [1] M. Olano, "Real-Time Rendering Techniques Using OpenGL," *ACM Transistion Graph.*, vol. 21, No.3, 2020.
- [2] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane, *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3.* in OpenGL. Pearson Education, 2013. [Online]. Available: <https://books.google.co.id/books?id=jG4LGmH5RuIC>
- [3] E. Angel and D. Shreiner, *Interactive computer graphics : a top-down approach using OpenGL*, 6. ed., Pearson international ed. Boston SE - 759 Seiten, [8] Blätter : Illustrationen, Diagramme: Pearson Addison-Wesley, 2012. doi: LK - <https://worldcat.org/title/741019867>.