VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590018

A Project Report

For the 6th semester

"ANALOG CLOCK"

Submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

Francis Dsouza	**2VD19CS012**
Jagadish S Munavalli	**2VD19CS013**

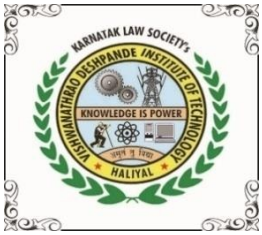
Under the Guidance of

Prof. JAYASHREE INCHAL

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERIKLS
VISHWANATHRAO DESHPANDE INSTITUTE OF TECHNOLOGY,
HALIYAL-581329

2021-2022

KLS VISHWANATHRAO DESHPANDE
INSTITUTE OF TECHNOLOGY, HALIYAL - 581329



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Certificate

Certified that the Project work entitled "**ANALOG CLOCK** **"** is bonafide work carried out by Francis dsouza (2VD19CS012),Jagadish S Munavalli (2VD19CS013) in partial fulfillment of the requirements for the award of the degree of **Bachelor of Computer Science and Engineering of Visvesvaraya Technological University,Belagavi,** during the year 2021-2022. It is certified that all the corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
Prof. Jayashree Inchal	**Prof. Poornima Raikar**	**Dr. V. A Kulkarni**

Name of the Examiners:	Signature with date:
1.	1.
2.	2.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

"Task successful" makes everyone happy. But the happiness will be gold without glitter if we didn't state the persons who have supported us to make it success. Success will be crowned to people who made it reality but the people whose constant guidance and encouragement made it possible will be crowned first on theeve of success.

We wish to express our sincere gratitude towards our guide Prof.Jayashree Inchal or her constant motivation and valuable help through the project.

We take this opportunity to thank Prof. Poornima Raikar HOD of Computer Science and Engineering Department for providing the inspiration for taking the project to its completion.

We are also grateful to our beloved principal Dr. V. A. Kulkarni for having provided us the academic environment, which nurtured our practical skills, contributing to the success of our project. We would like to thank all the teaching and non-teaching for their ideas and encouragement which helped us through the completion of the project.

We extend our sincere gratitude towards our parents who have encouraged us withtheir blessings to do this project successfully.

ABSTRACT

The project is idea to display the clock with computer graphics. This project just used the local time,fetching from the computer and display it on the screen. This project implements the wall clock with the round circular board and three conic geometric which forms the different(sec,min &hour)hands. Here we are going to use time function to display time. In this project we are going to get output has the current time

INDEX

List of Figure

CHAPTER 1

INTRODUCTION

The aim of this project is to implement an application package of Computer graphics using OpenGL. Here we represent the concepts displaying Analog clock in OpenGL.

Computer Graphics

Graphics provides one of the most natural means of communicating within a computer, since our highly developed 2D and 3D pattern-recognition abilities allow us to perceive and process pictorial data rapidly and effectively. Interactive computer graphics is the most important means of producing pictures since the invention of photography and television. It has the added advantage that, with the computer, we can make pictures not only of concrete real world objects but also of abstract, synthetic objects, such as mathematical surfaces and of data that have no inherent geometry, such as survey results.

Computer graphics started with the display of data on hardcopy plotters and cathode ray tube screens soon after the introduction of computers themselves. It has grown to include the creation, storage, and manipulation of models and images of objects. These models come from a diverse and expanding set of fields, and include physical, mathematical, engineering, architectural, and even conceptual structures, natural phenomena, and so on. Computer graphics today is largely interactive. The user controls the contents, structure, and appearance of the objects and of their displayed images by using input devices, such as keyboard, mouse, or touch-screen. Due to close relationships between the input devices and the display, the handling of such devices is included in the study of computer graphics. The advantages of the interactive graphics are many in number. Graphics provides one of the most natural means of communicating with a computer, since our highly developed 2D and 3D patter-recognition abilities allow us to perceive and process data rapidly and efficiently. In many design, implementation, and construction processes today, the information pictures can give is virtually indispensable. Scientific visualization

became an important field in the 1980s when the scientists and engineers realized that they could not interpret the prodigious quantities of data produced in supercomputer runs without summarizing the data and highlighting trends and phenomena in various kinds of graphical representations.

1.2 OpenGL Interface

OpenGL is an application program interface (API) offering various functions to implement primitives, models and images. This offers functions to create and manipulate render lighting, coloring, viewing the models. OpenGL offers different coordinate system and frames. OpenGL offers translation, rotation and scaling of objects. Most of our applications will be designed to access OpenGL directly through functions in three libraries. They are: 1. Main GL: Library has names that begin with the letter gl and are stored in a library usually referred to as GL. 2. OpenGL Utility Library (GLU): This library uses only GL functions but contains code for creating common objects and simplifying viewing. 3. OpenGL Utility Toolkit(GLUT): This provides the minimum functionality that should be accepted in any modern windowing system.

1.3 OpenGL Overview

- OpenGL (Open Graphics Library) is the interface between a graphic program and graphics hardware. It is streamlined. In other words, it provides low-level functionality. For example, all objects are built from points, lines and convex polygons. Higher level objects like cubes are implemented as six four-sided polygons.

- OpenGL supports features like 3-dimensions, lighting, anti-aliasing, shadows, textures, depth effects, etc.

- It is system-independent. It does not assume anything about hardware or operating system and is only concerned with efficiently rendering mathematically described scenes. As a result, it does not provide any windowing capabilities.

- It is a state machine. At any moment during the execution of a program there is a current model transformation

- It is a rendering pipeline. The rendering pipeline consists of the following steps:

o Defines objects mathematically.

o Arranges objects in space relative to a viewpoint.

o Calculates the color of the objects.

o Rasterize the objects.

Graphics provides one of the most natural means of communicating with a computer, since our highly developed 2D and 3D pattern-recognition abilities allow us to perceive and process pictorial data rapidly and efficiently. Interactive computer graphics is the most important means of producing pictures since the invention of photography and television. It has the added advantage that, with the computer, we can make pictures not only of concrete real world objects but also of abstract, synthetic objects, such as mathematical surfaces and of data that have no inherent geometry, such as survey results. OpenGL (open graphics library) is a standard specification defining a cross language cross platform API for writing applications that produce 2D and 3D computer graphics. OpenGL was developed by silicon graphics Inc. (SGI) in 1992 and is widely used in CAD, virtual reality, scientific visualization, information visualization and flight simulation. It is also used in video games. OpenGL serves two main purpose:

- To hide the complexities of interfacing with different 3D accelerators, by presenting programmer with a single, uniform API

- To hide the differing capabilities of hardware platforms, by requiring that all Implementations support the full openGL, feature set. OpenGL has historically been influential on the development of 3D accelerator, promoting a base level of functionality that is now common in consumer level hardware:

- Rasterized points, lines and polygons are basic primitives.

- A transform and lighting pipeline. Z buffering

- Texture Mapping.
- Alpha
- Blending

Chapter 2

SYSTEM REQUIREMENTS SPECIFICATION

2.1 HARDWARE REQUIREMENTS

The hardware requirements are very minimal and the software can run on most of the machines.

- Processor - Intel 486/Pentium processor or above.
- Processor Speed - 500 MHz or above.
- RAM - 64MB or above Storage Space - 2 MB or above, hard disk -10MB.
- Monitor resolution - A color monitor with a minimum resolution of 1000*700.
- Support both single & double buffering.

2.2 SOFTAWRE REQUIREMENTS

- An MS-DOS based operating system like Windows 98, Windows 2000 or WindowsXP, vista, windows 7 is the platform required to develop the 2D and 3D.
- A Visual C/C++ compiler is required for compiling the source code to make the executable file which can then be directly executed.
- A built in graphics library like glut and glut32, and header file like GL\glut.h and also dynamic link libraries like glut and glut32 are required for creating the 3D layout.

:FUNCTIONAL REQUIREMENTS:

OpenGL APIs:

If we want to have a control on the flow of program and if we want to interact with the window system then we use OpenGL API'S. Vertices are represented in the same manner internally, whether they are specified as two-dimensional or three-dimensional entities, everything that we do are here will be equally valid in three dimensions. Although OpenGL is easy to learn, compared with other APIs, it is nevertheless powerful. It supports the simple three dimensional programs and also supports the advanced rendering techniques.

GL/glut.h:

We use a readily available library called the OpenGL Utility Toolkit (GLUT), which provides the minimum functionality that should be expected in any modern windowing system.

The application program uses only GLUT functions and can be recompiled with the GLUT library for other window system. OpenGL makes a heavy use of macros to increase code readability and avoid the use of magic numbers. In most implementation, one of the include lines.

Chapter 3

LITERATURE SURVEY

The first digital pocket watch was the invention of Austrian engineer Josef Pallweber who created his "jump-hour" mechanism in 1883. Instead of a conventional dial, the jump-hour featured two windows in an enamel dial, through which the hours and minutes are visible on rotating discs. The second hand remained conventional. By 1885 Pallweber mechanism was already on the market in pocket watches by Cortébert and IWC; arguably contributing to the subsequent rise and commercial success of IWC. The principles of Pallweber jump-hour movement had appeared in wristwatches by the 1920s (Cortébert) and are still used today (Chronoswiss Digiteur). While the original inventor didn't have a watch brand at the time, his name has since been resurrected by a newly established watch manufacturer.

Plato clocks used a similar idea but a different layout. These spring-wound pieces consisted of a glass cylinder with a column inside, affixed to which were small digital cards with numbers printed on them, which flipped as time passed. The Plato clocks were introduced at the St. Louis World Fair in 1904, produced by Ansonia Clock Company. Eugene Fitch of New York patented the clock design in 1903. 13 years earlier Josef Pallweber had patented the same invention using digital cards (different from his 1885 patent using moving disks) in Germany (DRP No. 54093).The German factory Aktiengesellschaft für Uhrenfabrikation Lenzkirch made such digital clocks in 1893 and 1894.

Chapter 4

ABOUT THE PROJECT

Overview

Analog clocks usually indicate time using angles. The most common clock face uses a fixed numbered dial or dials and moving hand or hands. It usually has a circular scale of 12 hours , which can also serve as a scale of 60 minutes , and 60 seconds if the clock has a second hand. Many other styles and designs have been used throughout the years, including dials divided into 6, 8, 10, and 24 hours.

The only other widely used clock face today is the 24 hour analog dial , because of the use of 24 hour time in military organizations and timetables

User interface

- The background color of the clock is blue.
- Each solid cube is representing seconds.
- It contains 3 needles one for hour, one for minute, and one for seconds.
- Large needle represents minute, small needle represent hour and thin needle represents seconds.
- There is a proper delay between needle moments and timings.
- Needle moves from 1 to 12 in clockwise direction.

OBJECTIVE:

- This project shows the graphical representation of a working analogue clock.
- The objective of this program is to implement simple and basic functions of OpenGL.
- Computer graphics plays a major role in today's world where visualization takes the upper hand as compared to textual interaction.
- This is largely true as we can see user interface becoming more and more attractive all thanks to major leaps in the fields of computer graphics.
- The project is implemented using graphics OpenGL package provided by C.

CHAPTER 5

IMPLEMENTATION

EXISTING SYSTEM

The package which we have designed is a one which requires many graphics packages. Most of the functions which are needed to design the package are found in Turbo C compiler with a graphics package .The package requires simple in-built functions found in <GL/glut.h> library. This header file, in addition to the usual header files is needed for the working of the project. For running the program, any basic PC running compatible version of Microsoft Visual Studio is sufficient.

PROPOSED SYSTEM

To achieve three dimensional effects, open GL software is proposed. It is software which provides a graphical interface. It is an interface between application program and graphics hardware. The advantages are:

1. Open GL is designed as a streamlined.
2. It's a hardware independent interface i.e it can be implemented on many different hardware platforms.
3. With Open GL we can draw a small set of geometric primitives such as points, lines and polygons etc.
4. It provides double buffering which is vital in providing transformations.
5. It is event driven software.
6. It provides call back functions.

Function description:

This function includes glutTimerFunc it takes 3 Parameters.

- msecs- Number of milliseconds to pass before calling the callback.
- Func- The timer callback function.
- value- Integer value to pass to the timer callback.

glutTimerFunc registers the timer callback func to be triggered in at least msecs milliseconds. The value parameter to the timer callback will be the value of the value parameter to glutTimerFunc. Multiple timer callbacks at same or differing times may be registered simultaneously. The number of milliseconds is a lower bound on the time before the callback is generated. GLUT attempts to deliver the timer callback as soon as possible after the expiration of the callback's time interval. There is no support for canceling a registered callback. Instead, ignore a callback based on its value parameter when it is triggered.

OPENGL FUNCTIONS

glColor3f (float, float, float):-This function will set the current drawing color

glClear( ):-Takes a single argument that is the bitwise OR of several values indicating which buffer is to be cleared.

glClearColor ():-Specifies the red, green, blue, and alpha values used by **glClear** to clear the color buffers.

GlLoadIdentity( ):-the current matrix with the identity matrix.

glMatrixMode(mode):-Sets the current matrix mode, *mode* can be **GL_MODELVIEW, GL_PROJECTION or GL_TEXTURE.**

void glutInit (int *argc, charargv):-**Initializes GLUT, the arguments from main are passed in and can be used by the application.

void glutInitDisplayMode (unsigned int mode):-Requests a display with the properties in mode. The value of mode is determined by the logical OR of options including the color model and buffering.

void glutInitWindowSize (int width, int height):- Specifies the initial position of the top-left corner of the window in pixels

glutInitCreateWindow (char *title):-A window on the display.The string title can be used to label the window. The return value provides references to the window that can be used when there are multiple windows.

void glutMouseFunc(void *f(int button, int state, int x, int y):-Register the mouse callback function f. The callback function returns the button,the state of button after the event and the position of the mouse relative to the top-left corner of the window.

void glutKeyboardFunc(void(*func) (void)):-This function is called every time when you press enter key to resume the game or when you press ‘b’ or ‘B’ key to go back to the initial screen or when you press esc key to exit from the application.

void glutDisplayFunc (void (*func) (void)):-Register the display function func that is executed when the window needs to be redrawn.

void glutSpecialFunc(void(*func)(void)):-This function is called when you press the special keys in the keyboard like arrow keys, function keys etc. In our program, the func is invoked when the up arrow or down arrow key is pressed for selecting the options in the main menu and when the left or right arrow key is pressed for moving the object(car) accordingly.

glutPostReDisplay ( ) :-which requests that the display callback be executed after the current

callback returns.

void MouseFunc (void (*func) void)):-This function is invoked when mouse keys are pressed. This function is used as an alternative to the previous function i.e., it is used to move the object(car) to right or left in our program by clicking left and right button respectively.

void glutMainLoop ()

Cause the program to enter an event-processing loop.It should be the last statement in main function.

This function draws a analog clock. Displays clock face by using the OpenGL function gluDisk. Displays large wire cube using the OpenGL function gluWireCube.Displays hour hand,second hand and minute hand using the OpenGL function gluCylinder. Displays solid cube using the OpenGL function gluSolidCube which represents seconds.

CHAPTER 6

TESTING

Testing process started with the testing of individual program units such as functions or objects. These were then integrated into sub-systems and systems, and interactions of these units were tested.

Testing involves verification and validation.

Validation: “Are we building right product?”

Verification: “Are we building the product right?”

The ultimate goal of the verification and validation process is to establish confidence that the software system is ‘fit for purpose’. The level of required confidence depends on the system’s purpose, the expectations of the system users and the current marketing environment for the system.

With the verification and validation process, there are two complementary approaches to the system checking and analysis:

Software inspections or peer reviews analyses and check system representations such as the requirements document, design diagrams, and the program source code. Software testing involves running an implementation of the software with test data.

Testing process of this code is very simple. When we execute the program it displays the current system time properly during both day and night. It also displays proper digital time along with the date .

CHAPTER 7

SNAPSHOTS

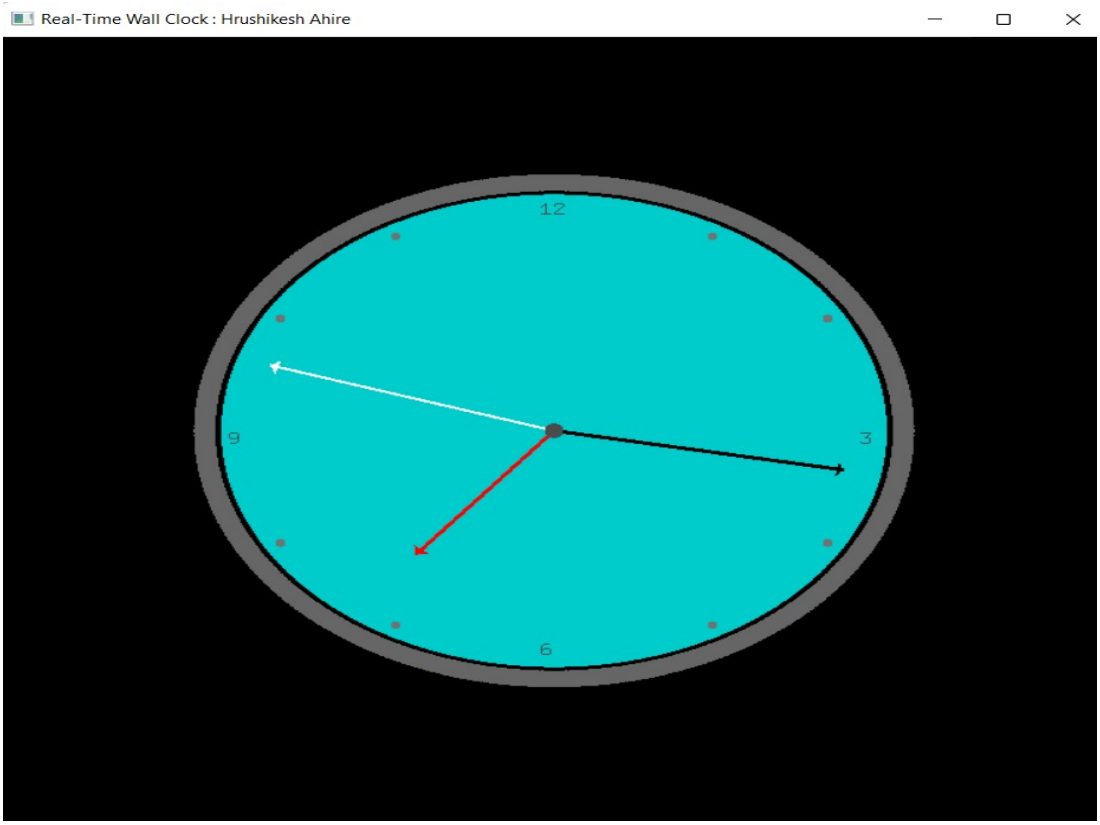


Fig1 : The analog clock

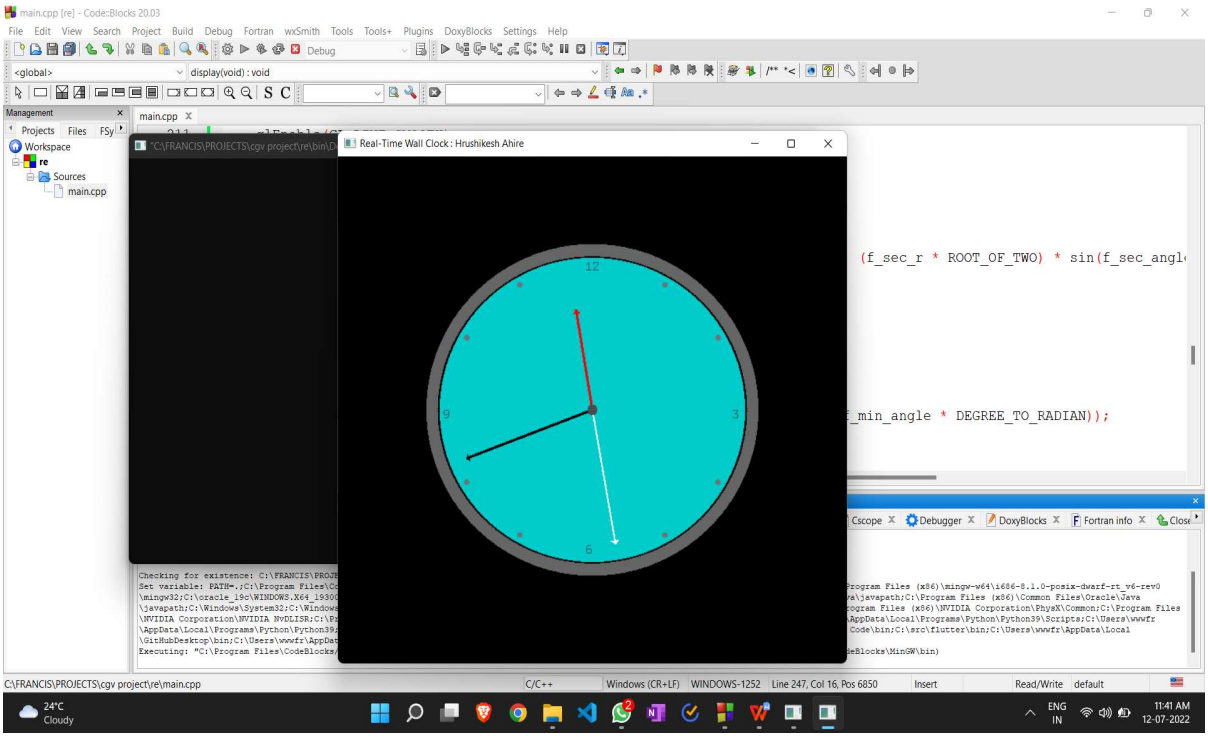


Fig2 : real time wall clock

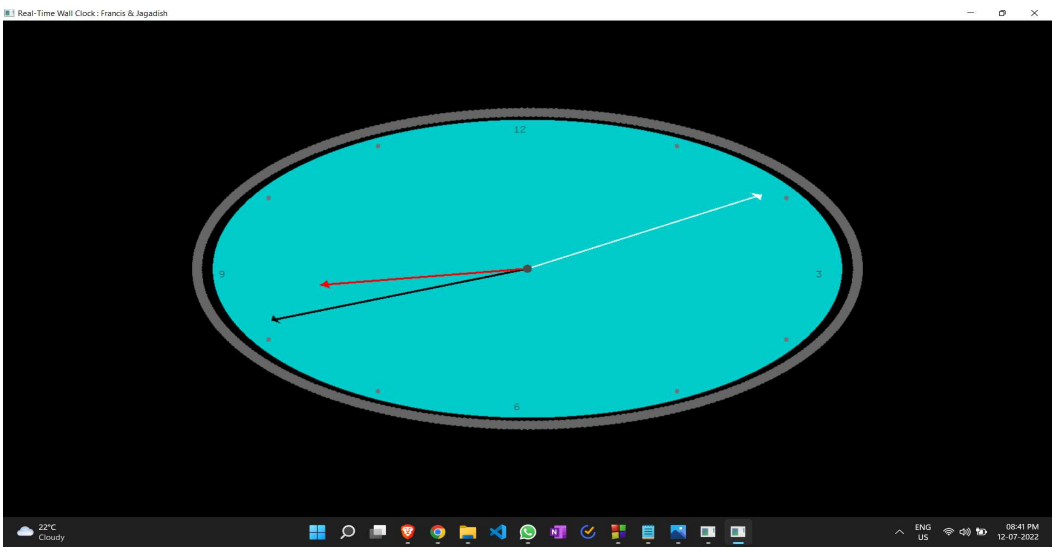


Fig3 : Full screen display of the digital clock

Chapter 8

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

This project shows the graphical representation of a working analogue clock. The objective of this program is to implement simple and basic functions of OpenGL. Computer graphics plays a major role in today's world where visualization takes the upper hand as compared to textual interaction. This is largely true as we can see user interface becoming more and more attractive all thanks to major leaps in the fields of computer graphics. The project is implemented using graphics OpenGL package provided by C.

The above argument is equally justified in the fields of computer simulation which involve complex graphics being highlighted at its peak. It is becoming more and more popular and the constant dive to improve particular system efficiency by studying its simulated model attracts more people towards it.

7.2 FUTURE ENHANCEMENTS

Though the package that is designed here does not include complex OpenGL package, we intend to improve this by including extra features like lighting effects, Changing the background color adding alarams. We do this with an aim to attract more people to use our package .

REFERENCES

[1] Edward Angel's Interactive Computer Graphics Pearson Education 5th Edition

[2] Interactive computer Graphics --A top down approach using open GL--by Edward Angle

[3] Jackie.L.Neider,Mark Warhol,Tom.R.Davis,"OpenGL Red Book",Second Revised Edition,2005.

[4] Donald D Hearn and M.Pauline Baker,"Computer Graphics with OpenGL", 3rd Edition.

[5] Portion of the code for implementing GEAR has been borrowed from Brian Paul's MESA.