

Computer Engineering Department

A.P. Shah Institute of Technology

G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year 2020-2021

Computer Graphics Mini-Project

2D-Animation : Windmill

Karthik Guttula (20102121)

Parth Mishra (20102162)

Sharvari Kasar (20102139)

Guided by: Prof. Shafaque Syed

Course Code: CSC305

1. Problem Definition

- To build a Graphical User Interface of a 2D animated 'Windmill' by implementing various concepts of Computer Graphics.
- We've used C++ programming language to build our animated windmill.

2. Hardware and Software Requirements

Hardware Requirements :

- ❖ Intel Pentium III 800 MHz Processor or higher version
- ❖ Intel chipset 810 mother board or higher version
- ❖ 14" color monitor or greater than that
- ❖ Mouse
- ❖ Keyboard
- ❖ 2GB HDD or greater
- ❖ 256 MB RAM or greater

Software Requirements :

- ❖ This project is built using C++ programming language.
- ❖ The software application used to build and compile this project is Turbo C++ as this compiler supports graphics.h package (special DOSBOXed installer for Turbo C++ compiler).

3. Functions/ Libraries/ Packages used in the Project

Libraries/headers used:-

(standard)

iostream.h - iostream.h is the header file which contains all the functions of program like cout, cin etc.

conio.h - conio.h is a C header file used mostly by MS-DOS compilers to provide console input/output. It is not part of the C standard library or ISO C, nor is it defined by POSIX.

stdio.h - The header file stdio. h stands for Standard Input Output. It has the information related to input/output functions.

graphics.h - graphics.h library is used to include and facilitate graphical operations in program. graphics.h functions can be used to draw different shapes, display text in different fonts, change colors and many more

math.h - math.h is a header file in the standard library of the C programming language designed for basic mathematical operations.

dos.h - dos.h is a header file in C Language. This library has functions that are used for handling like interrupts, producing sound, date and time functions, etc. It is Borland specific and works in compilers such as Turbo C Compiler.

Functions used:-

Pre-defined functions:

- line()** - to draw a line.
- kbhit()** - to stop the program by pressing any key. If a key has been pressed then it returns a non-zero value, otherwise a zero.
- cleardevice()** - to delete everything from the window at a particular instant.
- delay()** - to hold the current state for a particular time.
- closegraph()** - to close the graphics mode, deallocates all memory allocated by graphics system and restores the screen to the mode it was in before you called initgraph.

User-defined functions:

- void wind()** - to calculate new co-ordinates and after a rotation and draw the rotated triangle accordingly.

4. Algorithm of your Project

Step1: Start

Step2: Declare variables $x[7], y[7], maxx, maxy, xw1, yw1, xw2, yw2$

Step3: Enter value of $x[7], y[7], maxx, maxy$

Step4: Read the following co-ordinates to draw 2D figure and perform the following steps to achieve the necessary transformation

- ROTATION: Read the rotation angle and compute the new position using the formula

$$\text{➤ } xw1 = \cos(\theta) * X + \sin(\theta) * Y;$$

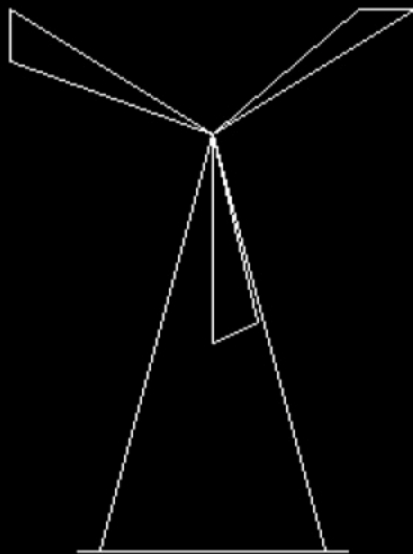
$$\text{➤ } yw1 = -\sin(\theta) * X + \cos(\theta) * Y;$$

and display new positions.

Step5: Stop.

5. Project Snapshots

Output



PROGRAM SNAPSHOTS

```
WINDMILL.CPP - Notepad
File Edit Format View Help
#include<iostream.h>

#include<conio.h>

#include<stdio.h>

#include<graphics.h>

#include<math.h>

#include<dos.h>

void wind(float x[7],float y[7]);

void main()
{
    int gd=DETECT,gm;

    float x[7],y[7],maxx,maxy,xw1,yw1,xw2,yw2;

    float theta=30;

    initgraph(&gd, &gm, "C:\\\\TURBOC3\\\\BGI");

    maxx=getmaxx();

    maxy=getmaxy();

    x[0]=maxx/2;

    y[0]=maxy/2;

    x[1]=y[4]=x[2]=-90;

    y[6]=y[5]=y[1]=60;

    y[2]=35;

    y[3]=-100;

    x[4]=20;

    x[3]=0;
```

P R O G R A M

```
WINDMILL.CPP - Notepad
File Edit Format View Help

x[3]=0;

x[5]=90;

x[6]=65;

theta=theta*22/7/180;

while(kbhit()==0)
{
    wind(x,y);

    xw1=cos(theta)*x[1]+sin(theta)*y[1];

    yw1=-sin(theta)*x[1]+cos(theta)*y[1];

    xw2=cos(theta)*x[2]+sin(theta)*y[2];

    yw2=-sin(theta)*x[2]+cos(theta)*y[2];

    x[1]=xw1;

    y[1]=yw1;

    x[2]=xw2;

    y[2]=yw2;

    xw1=cos(theta)*x[3]+sin(theta)*y[3];

    yw1=-sin(theta)*x[3]+cos(theta)*y[3];

    xw2=cos(theta)*x[4]+sin(theta)*y[4];

    yw2=-sin(theta)*x[4]+cos(theta)*y[4];

    x[3]=xw1;

    y[3]=yw1;

    x[4]=xw2;

    y[4]=yw2;

    xw1=cos(theta)*x[5]+sin(theta)*y[5];
```



```
WINDMILL.CPP - Notepad
File Edit Format View Help

xw1=cos(theta)*x[5]+sin(theta)*y[5];
yw1=-sin(theta)*x[5]+cos(theta)*y[5];
xw2=cos(theta)*x[6]+sin(theta)*y[6];
yw2=-sin(theta)*x[6]+cos(theta)*y[6];
x[5]=xw1;
y[5]=yw1;
x[6]=xw2;
y[6]=yw2;
delay(75);
cleardevice();
}
closegraph();
getch();
}
```

P R O G R A M

```
void wind(float x[7],float y[7])
{
    cleardevice();
    line(x[0],y[0],x[0]-50,y[0]+200);
    line(x[0],y[0],x[0]+50,y[0]+200);
    line(x[0]-60,y[0]+200,x[0]+60,y[0]+200);
    line(x[0],y[0],x[0]+x[1],y[0]-y[1]);
    line(x[0],y[0],x[0]+x[2],y[0]-y[2]);
    line(x[0]+x[1],y[0]-y[1],x[0]+x[2],y[0]-y[2]);
    line(x[0],y[0],x[0]+x[3],y[0]-y[3]);
    line(x[0],y[0],x[0]+x[4],y[0]-y[4]);
    line(x[0]+x[3],y[0]-y[3],x[0]+x[4],y[0]-y[4]);
    line(x[0],y[0],x[0]+x[5],y[0]-y[5]);
    line(x[0],y[0],x[0]+x[6],y[0]-y[6]);
    line(x[0]+x[5],y[0]-y[5],x[0]+x[6],y[0]-y[6]);
}
```

6. References

- <https://www.youtube.com/watch?v=j3Q4xglwxDQ>
- <https://math.hws.edu/graphicsbook/source/glut/opengl-cart-and-windmill-2d.c>
- <http://vardhamancse.yolasite.com/resources/Computer%20Graphics%20Lab%20Manual.pdf>
- <https://github.com/ImRohitSingh/Computer-Graphics/blob/master/WindMill.cpp>
- CG Lab Manual.

Thank You