ANALOG WALL CLOCK

Report submitted in partial fulfilment of the requirement for the degree of

Bachelor of Technology

In

Computer Science & Engineering

By

JIVITESH SABHARWAL

AND

JAYASHY SINHA

To

VIMAL GAUR

Maharaja Surajamal Institute of Technology

Affiliated to Guru Gobind Singh Indraprastha University

Janakpuri, New Delhi-58

Batch - CSE-1

INTRODUCTION - ANALOG WALL CLOCK

A **clock** is a device used to measure, keep, and indicate time. The clock is one of the oldest human inventions, meeting the need to measure intervals of time shorter than the natural units: the day, the lunar month, and the year. Devices operating on several physical processes have been used over the millennia.

Analog clocks use angles to tell time. They have hands that rotate around the clock's face. The position of the hands shows the time. The face of the clock is a flat disk. It will often have the numbers one through twelve on the face to make it easier to read. Analog clocks commonly have two or three hands. If it has two, there is a large hand or minute hand and a smaller hand, the hour hand. Clocks with three hands also have a second hand. This hand is usually about as long as the minute hand, but much thinner.

Each hand shows the time it is named for and moves around the face of the clock one complete rotation for each movement to the next larger hand. For example, the second hand moves around the face of the clock in 60 seconds. It moves once each second. When it moves all the way around the clock, the minute hand moves forward one space. When the minute hand moves all the way around the face of the clock (which takes 60 minutes), the hour hand moves forward one section. The second and minute hands take 60 movements to move all around the face of the clock, the hour hand only needs 12 movements to do the same.

CODE

```
#include<conio.h>

#include<iostream.h>

#include<graphics.h>

#include<dos.h>

#include<math.h>

#include<stdlib.h>

#include<alloc.h>

void main()

{

int d=DETECT,m,r=50;

float gx,gy,x,y,x1,y1,xm,ym,xh,yh,k=0,i,j,ii,l;

initgraph(&d,&m,"c:\\turboc3\\bgi");

struct time t;

void *mem;

int size;

ii=13.89;

setcolor(14);

line(159,100,159,387);

line(450,100,450,382);

line(159,100,450,100);

setcolor(14);

arc(300,100,236,307,250);

arc(300,100,244,298,320);
```

3

```
//14.36,13.89

while(1)

{

while(ii<14.36)

 {

 if(kbhit())

   exit(1);

 // second's pixel

 for(i=11;i<=17.3;i+=0.105)

  {

  gx=55*cos(i);

  gy=55*sin(i);

  putpixel(gx+300,gy+200,4);


  }

 // outer circle

    setcolor(12);

    circle(300,200,85);

    circle(300,200,87);

    circle(300,200,90);

    setcolor(15);

    circle(300,200,80);

    setcolor(4);

    outtextxy(330,143,"1");
```

```
    outtextxy(350,165,"2");

    outtextxy(360,197,"3");

    outtextxy(352,228,"4");

    outtextxy(328,252,"5");

    outtextxy(298,260,"6");

    outtextxy(264,252,"7");

    outtextxy(242,226,"8");

    outtextxy(235,197,"9");

    outtextxy(235,165,"10");

    outtextxy(260,140,"11");

    outtextxy(292,135,"12");
//time function
///*
    gettime(&t);

    gotoxy(35,20);

    cout<<int(t.ti_hour)<<":"<<int(t.ti_min)<<":"<<int(t.ti_sec)<<"";

    i=(int(t.ti_sec)*(0.105))+11;

    j=(int(t.ti_min)*(0.105))+11;

    k=((int(t.ti_hour)*(0.105))*5+11);


    int min=int (t.ti_min);

    int rem=min/12;

    k=k+(rem*0.105);
///*/
```

```
//calculations for second hand

  x=r*cos(i);

  y=r*sin(i);

  setcolor(14);

  line(300,200,x+300,y+200);


//calculations for minute hand

  xm=(r-5)*cos(j);

  ym=(r-5)*sin(j);

  setcolor(9);

  line(300,200,xm+300,ym+200);


 //calculations for hour hand

  xh=(r-20)*cos(k);

  yh=(r-20)*sin(k);

  setcolor(6);

  line(300,200,xh+300,yh+200);


 delay(200.5);


  int pks=0;


if(kbhit())
```

```
exit(1);

float l,m;

 for(pks=1;pks<=4;pks++)

 {


 l=200*cos(ii);

 m=200*sin(ii);


 setcolor(4);

 line(300,290,300+l,190+m);


 setcolor(14);

 setfillstyle(SOLID_FILL,14);

 circle(l+300,m+190,15);

 floodfill(l+300,m+190,14);


 ii+=.105;


 delay(200.5);

 setcolor(0);

 setfillstyle(SOLID_FILL,0);

 circle(l+300,m+190,15);

 floodfill(l+300,m+190,0);
```

```
    line(300,290,300+l,190+m);

  }


  setcolor(0);
line(300,200,x+300,y+200);

  line(300,200,xm+300,ym+200);

  line(300,200,xh+300,yh+200);

}


  while(ii>13.89)

  {

  if(kbhit())

   exit(1);


 for(i=11;i<=17.3;i+=0.105)

  {

  gx=55*cos(i);

  gy=55*sin(i);

  putpixel(gx+300,gy+200,4);

  }
        setcolor(12);

            circle(300,200,85);

            circle(300,200,87);
```

```
       circle(300,200,90);

  setcolor(15);

setfillstyle(SOLID_FILL,0);

 circle(300,200,80);

floodfill(300,200,0);

 setcolor(4);

 outtextxy(330,143,"1");

 outtextxy(350,165,"2");

 outtextxy(360,197,"3");

 outtextxy(352,228,"4");

 outtextxy(328,252,"5");

 outtextxy(298,260,"6");

 outtextxy(264,252,"7");

 outtextxy(242,226,"8");

 outtextxy(235,197,"9");

 outtextxy(235,165,"10");

 outtextxy(260,140,"11");

 outtextxy(292,135,"12");

 gettime(&t);

 gotoxy(35,20);

 cout<<int(t.ti_hour)<<":"<<int(t.ti_min)<<":"<<int(t.ti_sec)<<"";

 i=(int(t.ti_sec)*(0.105))+11;

 j=(int(t.ti_min)*(0.105))+11;
```

```
k=((int(t.ti_hour)*(0.105))*5+11);

 int min=int (t.ti_min);

 int rem=min/12;

k=k+(rem*0.105);


 x=r*cos(i);

 y=r*sin(i);

 setcolor(14);

line(300,200,x+300,y+200);


 xm=(r-5)*cos(j);

 ym=(r-5)*sin(j);

 setcolor(9);

line(300,200,xm+300,ym+200);


 xh=(r-20)*cos(k);

 yh=(r-20)*sin(k);

 setcolor(6);

line(300,200,xh+300,yh+200);

 delay(200.5);



if(kbhit())

 exit(1);
```

```
 for(int pks=1;pks<=4;pks++)

 {

l=200*cos(ii);

m=200*sin(ii);


 setcolor(4);

line(300,290,300+l,190+m);

 setcolor(14);

 setfillstyle(SOLID_FILL,14);

 circle(l+300,m+190,15);

 floodfill(l+300,m+190,14);


 ii-=0.105;

delay(200.5);


 setcolor(0);

 setfillstyle(SOLID_FILL,0);

 circle(l+300,m+190,15);

 floodfill(l+300,m+190,0);

 line(300,290,300+l,190+m);

 }
setcolor(0);

 line(300,200,x+300,y+200);

 line(300,200,xm+300,ym+200);
```

```
      line(300,200,xh+300,yh+200);

  }

  }

}
```

RESULT AND DISCUSSION

Library Function used in the program-

1. Line () - line function is used to draw a line from a point(x1,y1) to point(x2,y2) i.e. (x1,y1) and (x2,y2) are end points of the line.

2. Setcolor() - In Turbo Graphics each color is assigned a number. Total 16 colors are available. Strictly speaking number of available colors depends on current graphics mode and driver.

3. Arc() - "arc" function is used to draw an arc with center (x, y) and start angle specifies starting angle, end angle specifies the end angle and last parameter specifies the radius of the arc.

4. Kbhit() - Function kbhit in C is used to determine if a key has been pressed or not. To use it in a program you should include the header file "conio. h". If a key has been pressed, then it returns a non zero value otherwise it returns zero.

5. Putpixel() - putpixel function plots a pixel at location (x, y) of specified color.

6. Circle() - Circle function is used to draw a circle with center (x,y) and third parameter specifies the radius of the circle. The code given below draws a circle.

7. Gettime() - gettime function is used to find current system time. We pass address of a structure varibale of type (struct time).

8. Gotoxy() - gotoxy function places cursor at a desired location on screen i.e., we can change cursor position using gotoxy function.

9. Setfillstyle() - setfillstyle function sets the current fill pattern and fill color.

10. Floodfill() - floodfill function is used to fill an enclosed area. Current fill pattern and fill.

11. Outtextxy() - outtextxy function display text or string at a specified point(x,y) on the screen.

20:38:11

20:38:17

20:38:23
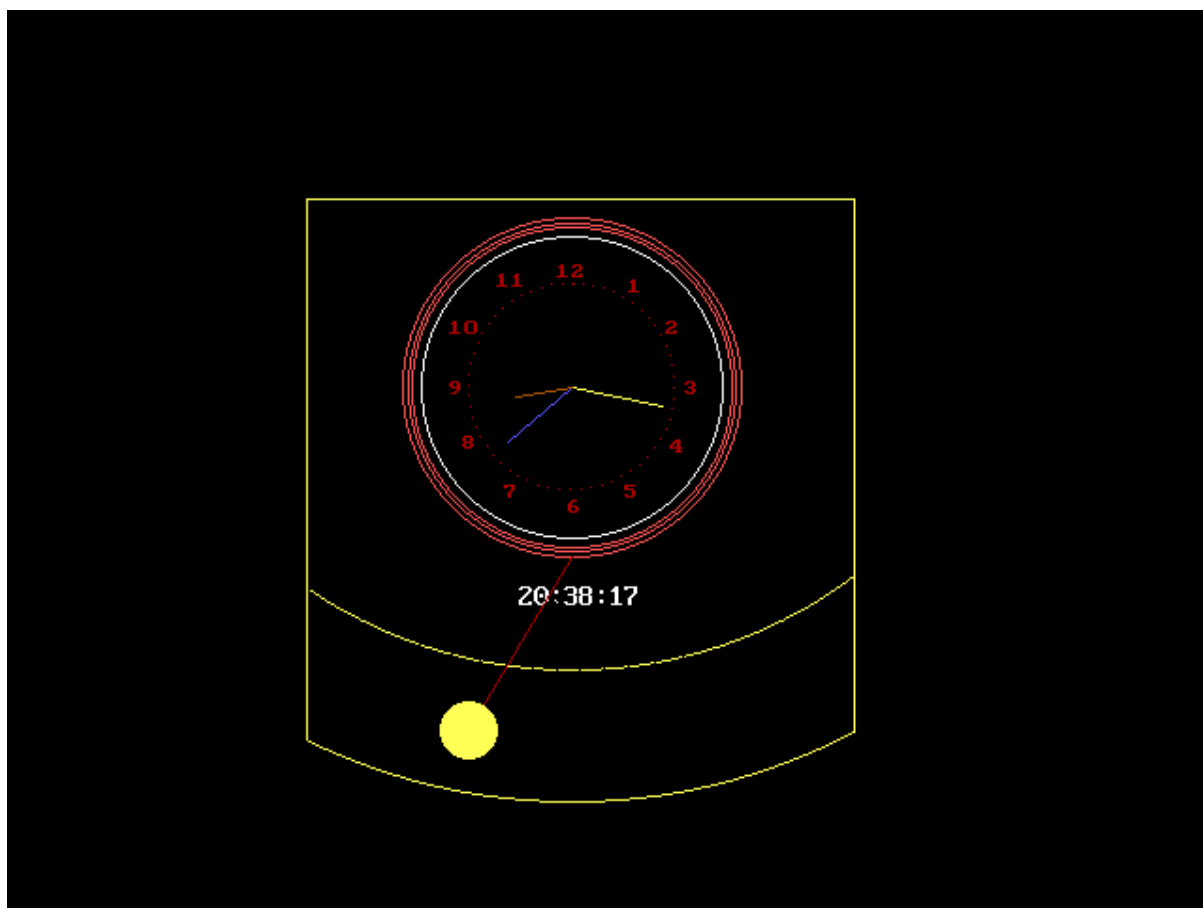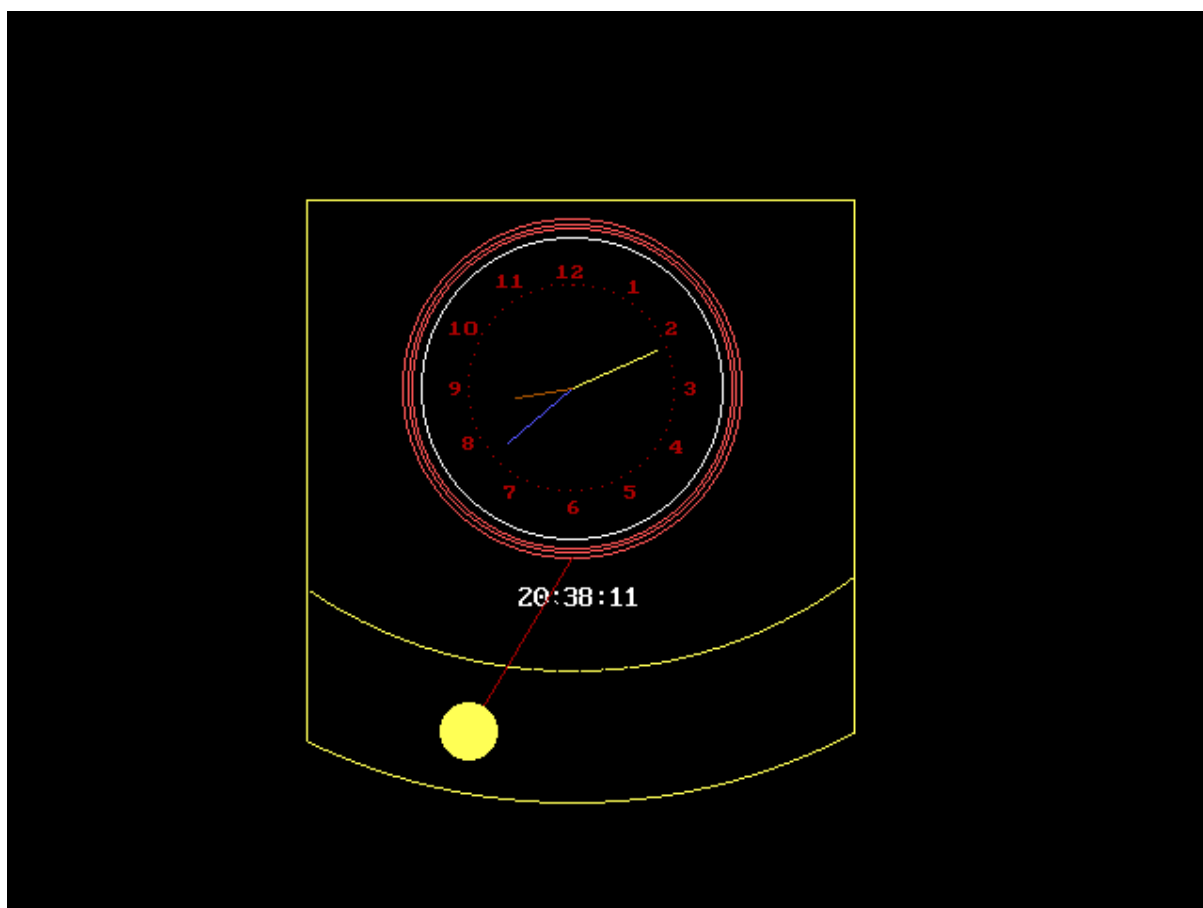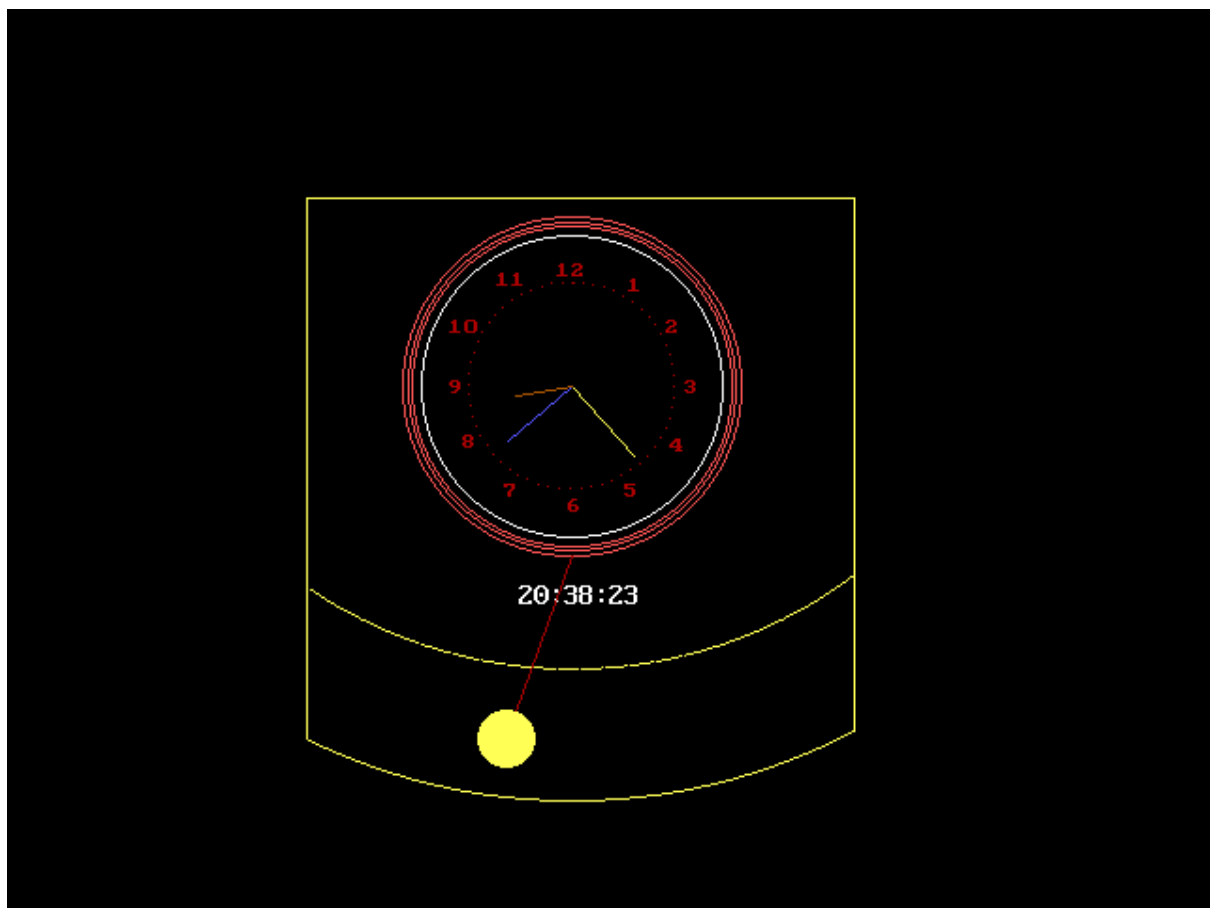
CONCLUSION

1. This is a computer graphics mini project aimed at designing the Analog Clock using C++ and basics of C language.

2. The Implementation Program for analog clock and Snapshots of the Output of this Implementation are provided above.

FUTURE SCOPE

1. It is Interactive 2D graphics.
2. The clock will be displayed.
3. Time will be displayed as same as system time.

REFERENCES

Books-

1. Donald Hearn and M.Pauline Baker, —Computer Graphics C version‖, Second Edition, Pearson Education.
2. Ralf Steinmetz & Klara Nahrstedt, —Multimedia Computing Communication & Applications‖, Pearson Education.

Links-

   a. www.geeksforgeeks.com
   b. www.stackoverflow.com
   c. www.wikipedia.com
   d. https://www.tutorialspoint.com/
   e. https://www.javatpoint.com/