MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Skill Based Mini Project Report

on

MAKING ANALOG CLOCK USING CPP

Submitted by:

Amit sharma

0901CA211006

Mentor:

Dr. Anshu Chaturvedi, professor

Submitted to:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

JULY-DEC 2021

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Amit sharma** (0901CA211006) has submitted the project report titled **Analog clock using c++** under the mentorship of **Dr. Anshu Chaturvedi (Professor),** as the skill based mini project in 1st year of Master of Computer Application of Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

Dr. Anshu Chaturvedi
Professor
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the fulfilment of partial requirement for the skill based mini project in 1st of Master of Computer Application in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Dr. Anshu Chaturvedi**, (**professor),** MITS Gwalior

I declare that I have not submitted the matter embodied in this report anywhere else.

Amit sharma
0901CA211006
1 Year,
Master of Computer Application,
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary project. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering, for allowing** me to explore this project. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty coordinator. I am grateful to the guidance of **Dr. Anshu Chaturvedi**, (professor)**,** computer science and engineering, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Amit sharma
0901CA211006
1 Year,
Master of Computer Application,
Computer Science and Engineering

ABSTRACT

Analog clock using cpp is just made by setting the pixels on x and y axis correctly that our clock looks as we want and we have to set the right timing that how our needles works that we made from setting pixels to made a line and we have to set the delay that our clock work accurately as original analog clock work. We made a circle shape clock and set numbers on it and made 3 needles the hour hand, the minute hand and the second hand and set the delay of how our 3 hands are moving.

TABLE OF CONTENTS

INTRODUCTION

Analog clock is made using cpp language and use graphics.h header file for using graphics functions. graphics.h can be used to draw different shapes, display text in different fonts, change colours and many more. Using functions of graphics.h in Turbo C compiler you can make graphics programs, animations, projects, and games. You can draw circles, lines, rectangles, bars and many other geometrical figures. You can change their colours using the available functions and fill them. Graphics.h gives us amazing functions like – circle(), putpixel(), setcolor() etc. that we can use to made graphics program in cpp.

I used these functions to set the pixels at right position I made a circle and put numbers in the circle and set a point on the centre and made 3 needles on it the hour hand, the minute hand and the second hand and set the delay or the second hand at 1 second and minute hand at 60 seconds and the hour hand at 60 minutes.

OBJECTIVE

My main objective to make this analog clock using cpp is to improve my concepts in cpp language and to learn graphics using cpp. And want to practice graphics.h functions like circle(), putpixel(), setcolor() and many more and also have to learn how to set pixels on right place that we want on x and y axis.

And by making this project I am able to revise my all cpp concepts like loops variables etc. and also revised and learn graphics functions.

CODE

clock.cpp - Code::Blocks 20.03

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

<global>

Management

Projects Files FSy

Workspace

clock.cpp

```
#include <graphics.h>
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include <conio.h>
#include<math.h>
#include<dos.h>
#include<string.h>
#include<iostream>
#include<ctime>



#define S_N_L (radius-10)        // Second Needle Length
#define S_N_C RED                // Second needle Color
#define M_N_L (radius-20)        // Minute Needle Length
#define M_N_C LIGHTRED           // Minute Needle Color
#define H_N_L (radius-(radius/2))       // Hour Needle Length
#define H_N_C CYAN               // Hour Needle Color

float cx,cy;
float radius=100;

void draw_face(float radius);
void get_time(int &h,int &m,int &s);
void second_needle(int s);
void minute_needle(int m,int s);
void hour_needle(int h,int m,int s);

int main(void)
{
/* request auto detection */
int gdriver = DETECT, gmode, errorcode;
initgraph(&gdriver,&gmode,"d:\\tc\\bgi");
/*********************************/
cx=getmaxx()/2.0; // cx is center x value.
cy=getmaxy()/2.0; // cy is center y value.
/** Now the point (cx,cy) is the center of your screen. **/
```

clock.cpp - Code::Blocks 20.03

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

<global>

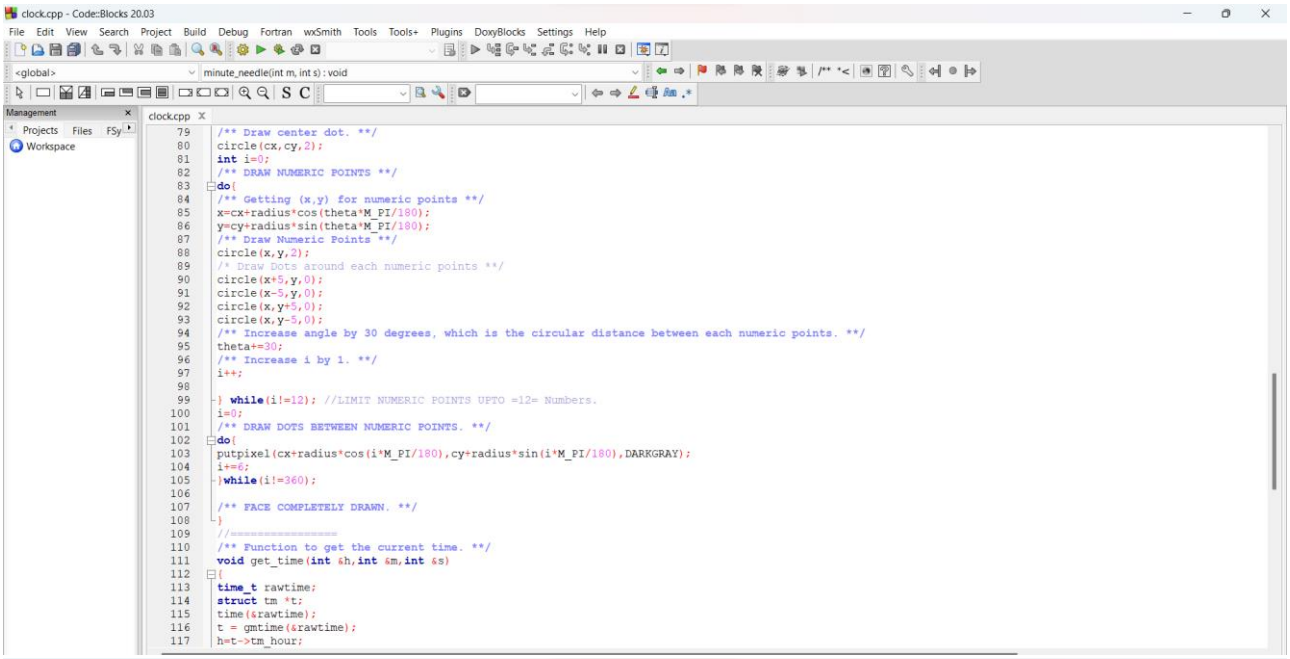
Management

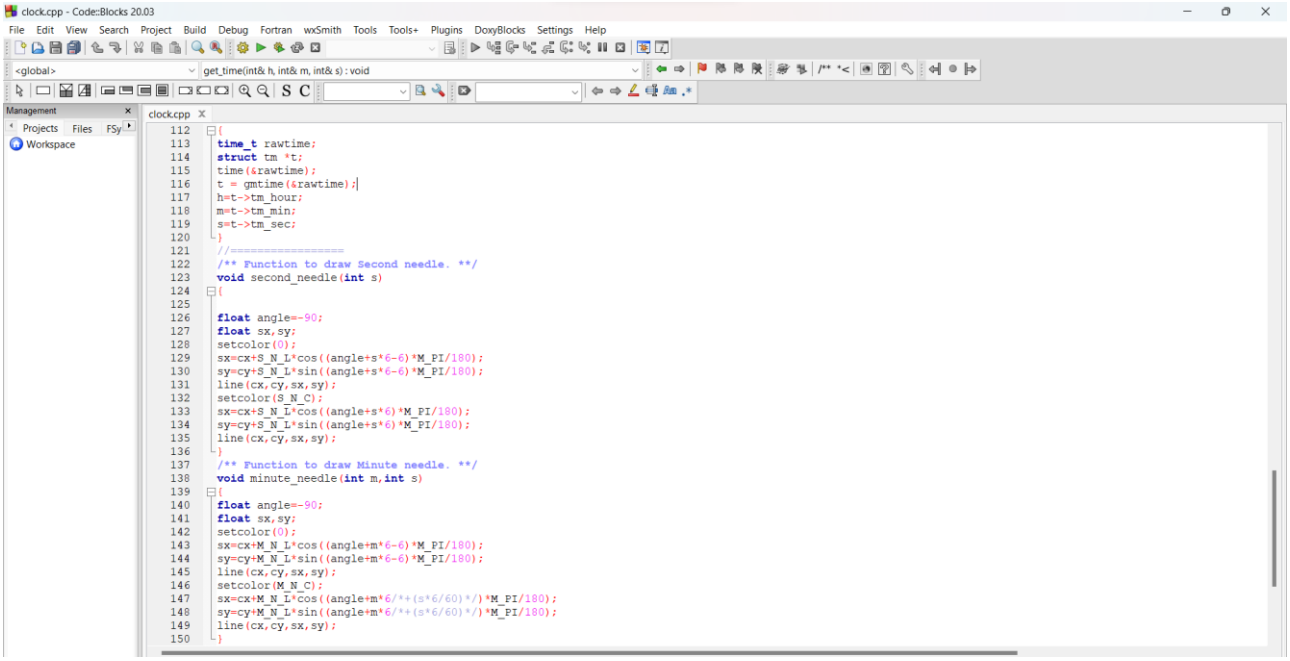
Projects Files FSy

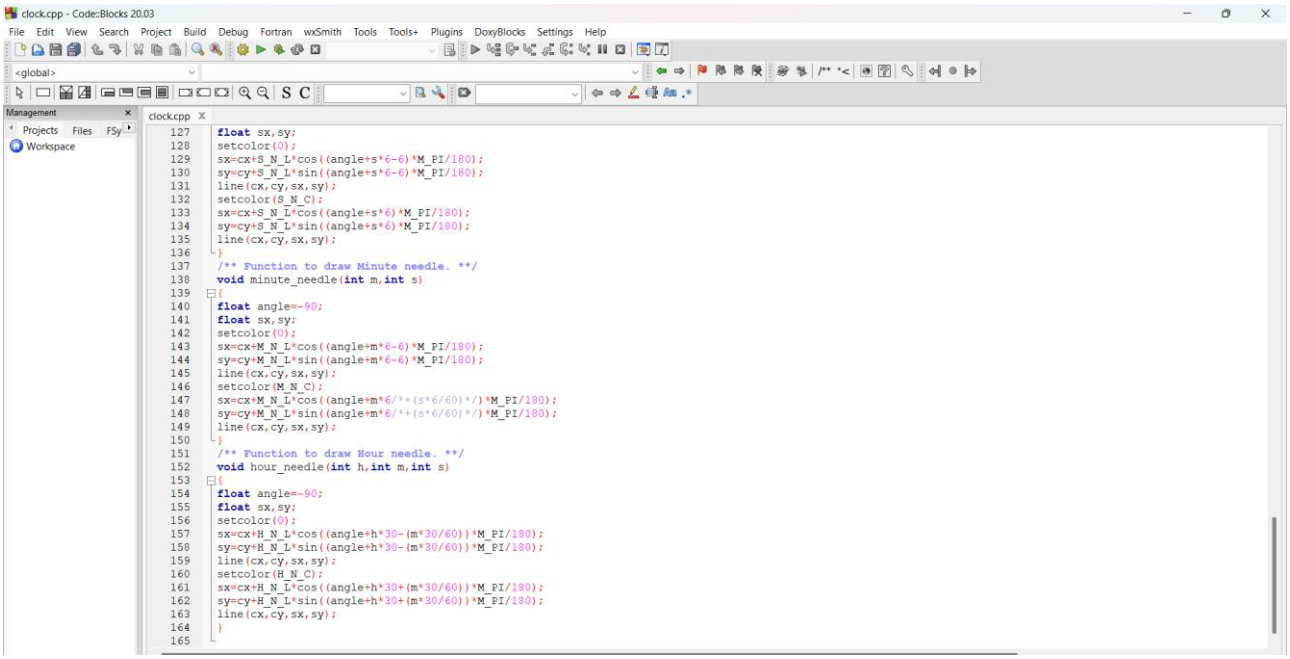
Workspace

clock.cpp

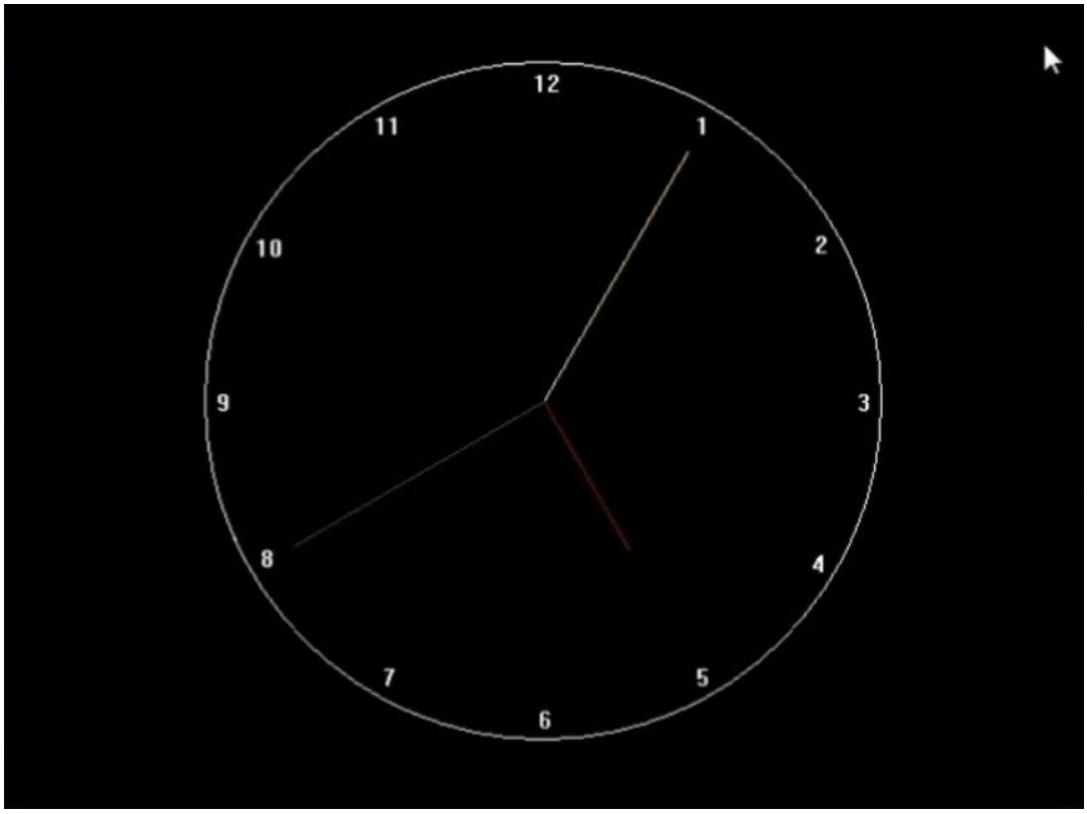
```
float x,y;
int hour,minute,second;
draw_face(radius);
while(!kbhit())
{
get_time(hour,minute,second);
second_needle(second);
minute_needle(minute,second);
hour_needle(hour,minute,second);
circle(cx,cy,2);
delay(100);
}


getch();
closegraph();
return 0;

}
//*************** FUNCTIONS DEFINITIONS ****************//

void draw_face(float radius)
{

int theta=0; // theta is the angle variable.
float x,y;
/** Draw Clock Border. **/
circle(cx,cy,radius+24);
circle(cx,cy,radius+23);
/** Draw GREEN material border. **/
setcolor(BROWN);    // I like a wooden frame!
/** Paint the border. **/
for(int i=0;i<9;i++)
circle(cx,cy,radius+13+i);
/** Set the color white. **/
setcolor(WHITE);
/** Draw outer-inner border. **/
circle(cx,cy,radius+12);
circle(cx,cy,radius+10);
```


clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
minute_needle(int m, int s) : void
Management
Projects Files FSy
Workspace
clock.cpp
/** Draw center dot. **/
circle(cx,cy,2);
int i=0;
/** DRAW NUMERIC POINTS **/
do{
/** Getting (x,y) for numeric points **/
x=cx+radius*cos(theta*M_PI/180);
y=cy+radius*sin(theta*M_PI/180);
/** Draw Numeric Points **/
circle(x,y,2);
/* Draw Dots around each numeric points **/
circle(x+5,y,0);
circle(x-5,y,0);
circle(x,y+5,0);
circle(x,y-5,0);
/** Increase angle by 30 degrees, which is the circular distance between each numeric points. **/
theta+=30;
/** Increase i by 1. **/
i++;
} while(i!=12); //LIMIT NUMERIC POINTS UPTO =12= Numbers.
i=0;
/** DRAW DOTS BETWEEN NUMERIC POINTS. **/
do{
putpixel(cx+radius*cos(i*M_PI/180),cy+radius*sin(i*M_PI/180),DARKGRAY);
i+=6;
}while(i!=360);
/** FACE COMPLETELY DRAWN. **/
}
//================
/** Function to get the current time. **/
void get_time(int &h,int &m,int &s)
{
time_t rawtime;
struct tm *t;
time(&rawtime);
t = gmtime(&rawtime);
h=t->tm_hour;


clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
get_time(int& h, int& m, int& s) : void
Management
Projects Files FSy
Workspace
clock.cpp
{
time_t rawtime;
struct tm *t;
time(&rawtime);
t = gmtime(&rawtime);
h=t->tm_hour;
m=t->tm_min;
s=t->tm_sec;
}
//================
/** Function to draw Second needle. **/
void second_needle(int s)
{
float angle=-90;
float sx,sy;
setcolor(0);
sx=cx+S_N_L*cos((angle+s*6-6)*M_PI/180);
sy=cy+S_N_L*sin((angle+s*6-6)*M_PI/180);
line(cx,cy,sx,sy);
setcolor(S_N_C);
sx=cx+S_N_L*cos((angle+s*6)*M_PI/180);
sy=cy+S_N_L*sin((angle+s*6)*M_PI/180);
line(cx,cy,sx,sy);
}
/** Function to draw Minute needle. **/
void minute_needle(int m,int s)
{
float angle=-90;
float sx,sy;
setcolor(0);
sx=cx+M_N_L*cos((angle+m*6-6)*M_PI/180);
sy=cy+M_N_L*sin((angle+m*6-6)*M_PI/180);
line(cx,cy,sx,sy);
setcolor(M_N_C);
sx=cx+M_N_L*cos((angle+m*6/*+(s*6/60)*/)*M_PI/180);
sy=cy+M_N_L*sin((angle+m*6/*+(s*6/60)*/)*M_PI/180);
line(cx,cy,sx,sy);
}


clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
Management
Projects Files FSy
Workspace
clock.cpp
127 float sx,sy;
128 setcolor(0);
129 sx=cx+S_N_L*cos((angle+s*6-6)*M_PI/180);
130 sy=cy+S_N_L*sin((angle+s*6-6)*M_PI/180);
131 line(cx,cy,sx,sy);
132 setcolor(S_N_C);
133 sx=cx+S_N_L*cos((angle+s*6)*M_PI/180);
134 sy=cy+S_N_L*sin((angle+s*6)*M_PI/180);
135 line(cx,cy,sx,sy);
136 }
137 /** Function to draw Minute needle. **/
138 void minute_needle(int m,int s)
139 {
140 float angle=-90;
141 float sx,sy;
142 setcolor(0);
143 sx=cx+M_N_L*cos((angle+m*6-6)*M_PI/180);
144 sy=cy+M_N_L*sin((angle+m*6-6)*M_PI/180);
145 line(cx,cy,sx,sy);
146 setcolor(M_N_C);
147 sx=cx+M_N_L*cos((angle+m*6/**+(s*6/60)*/)*M_PI/180);
148 sy=cy+M_N_L*sin((angle+m*6/**+(s*6/60)*/)*M_PI/180);
149 line(cx,cy,sx,sy);
150 }
151 /** Function to draw Hour needle. **/
152 void hour_needle(int h,int m,int s)
153 {
154 float angle=-90;
155 float sx,sy;
156 setcolor(0);
157 sx=cx+H_N_L*cos((angle+h*30-(m*30/60))*M_PI/180);
158 sy=cy+H_N_L*sin((angle+h*30-(m*30/60))*M_PI/180);
159 line(cx,cy,sx,sy);
160 setcolor(H_N_C);
161 sx=cx+H_N_L*cos((angle+h*30+(m*30/60))*M_PI/180);
162 sy=cy+H_N_L*sin((angle+h*30+(m*30/60))*M_PI/180);
163 line(cx,cy,sx,sy);
164 }
165

OUTPUT



CONCLUSION

The project is complete and in last I have accomplished my all objectives.

And I have learned and practice all concepts that I want.

But in future I am going to add new things in this project for better improvements and try to learn more new things by applying in this project.

REFRENCES

During my learning phase for making this project I have used these references:

1. Let us C by Yashwant kanetkar
2. https://www.geeksforgeeks.org/c-plus-plus/?ref=shm
3. https://www.w3schools.com/cpp/default.asp