

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:

Aman Shakya

0901CA211005

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 **Aman Shakya** (0901CA211005) 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

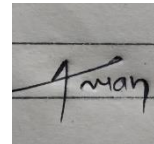
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(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.



Aman Shakya
0901CA211005

1 Year,
Master of Computer Application,
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

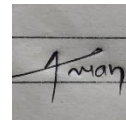
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

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.



Aman Shaky

0901CA211005

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

TITLE	PAGE NO.
Introduction	1
Objective	2
Code	3
Output	6
Conclusion	7
References	8

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 DoryBlocks Settings Help
<global>
Management
  Projects Files FSy
  Workspace
clock.cpp X
1 #include <graphics.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <stdio.h>
5 #include <conio.h>
6 #include <math.h>
7 #include <dos.h>
8 #include <string.h>
9 #include <iostream>
10 #include <ctime>
11
12
13
14 #define S_N_L (radius-10) // Second Needle Length
15 #define S_N_C RED // Second Needle Color
16 #define M_N_L (radius-20) // Minute Needle Length
17 #define M_N_C LIGHTRED // Minute Needle Color
18 #define H_N_L (radius-(radius/2)) // Hour Needle Length
19 #define H_N_C CYAN // Hour Needle Color
20
21 float cx,cy;
22 float radius=100;
23
24 void draw_face(float radius);
25 void get_time(int sh,int sm,int ss);
26 void second_needle(int s);
27 void minute_needle(int m,int s);
28 void hour_needle(int h,int m,int s);
29
30 int main(void)
31 {
32     /* request auto detection */
33     int gdriver = DETECT, gmode, errorcode;
34     initgraph(&gdriver,&gmode,"d:\\tc\\bgi");
35     /******
36     cx=getmaxx()/2.0; // cx is center x value.
37     cy=getmaxy()/2.0; // cy is center y value.
38     /* Now the point (cx,cy) is the center of your screen. */
39
40 float x,y;
41 int hour,minute,second;
42 draw_face(radius);
43 while(!kbhit())
44 {
45     get_time(hour,minute,second);
46     second_needle(second);
47     minute_needle(minute,second);
48     hour_needle(hour,minute,second);
49     circle(cx,cy,2);
50     delay(100);
51 }
52
53 getch();
54 closegraph();
55 return 0;
56 }
57
58 /****** FUNCTIONS DEFINITIONS *****/
59
60 void draw_face(float radius)
61 {
62     int theta=0; // theta is the angle variable.
63     float x,y;
64     /* Draw Clock Border. */
65     circle(cx,cy,radius+24);
66     circle(cx,cy,radius+23);
67     /* Draw GREEN material border. */
68     setcolor(BROWN); // I like a wooden frame!
69     /* Paint the border. */
70     for(int i=0;i<9;i++)
71     circle(cx,cy,radius+3+i);
72     /* Set the color white. */
73     setcolor(WHITE);
74     /* Draw outer-inner border. */
75     circle(cx,cy,radius+2);
76     circle(cx,cy,radius+3);
```

```

clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoryBlocks Settings Help
- global -
minute_needle(int m, int s): void
79 /** Draw center dot. */
80 circle(cx,cy,2);
81 int i=0;
82 /** DRAW NUMERIC POINTS */
83 do{
84 /** Getting (x,y) for numeric points */
85 x=cx+radius*cos(theta*M_PI/180);
86 y=cy+radius*sin(theta*M_PI/180);
87 /** Draw Numeric Points */
88 circle(x,y,2);
89 /** Draw dots around each numeric points */
90 circle(x+5,y,5);
91 circle(x-5,y,5);
92 circle(x,y+5,5);
93 circle(x,y-5,5);
94 /** Increase angle by 30 degrees, which is the circular distance between each numeric points. */
95 theta+=30;
96 /** Increase i by 1. */
97 i++;
98 } while(i!=12); //LIMIT NUMERIC POINTS UPTO =12= Numbers.
99 i=0;
100 /** DRAW DOTS BETWEEN NUMERIC POINTS. */
101 do{
102 putpixel(cx+radius*cos(i*M_PI/180),cy+radius*sin(i*M_PI/180),DARKGRAY);
103 i+=6;
104 } while(i!=360);
105 /** FACE COMPLETELY DRAWN. */
106 }
107 //=====
108 /** Function to get the current time. */
109 void get_time(int &h,int &m,int &s)
110 {
111 time_t rawtime;
112 struct tm *t;
113 time(&rawtime);
114 t = gmtime(&rawtime);
115 h=t->tm_hour;
116 m=t->tm_min;
117 s=t->tm_sec;
118 }

```

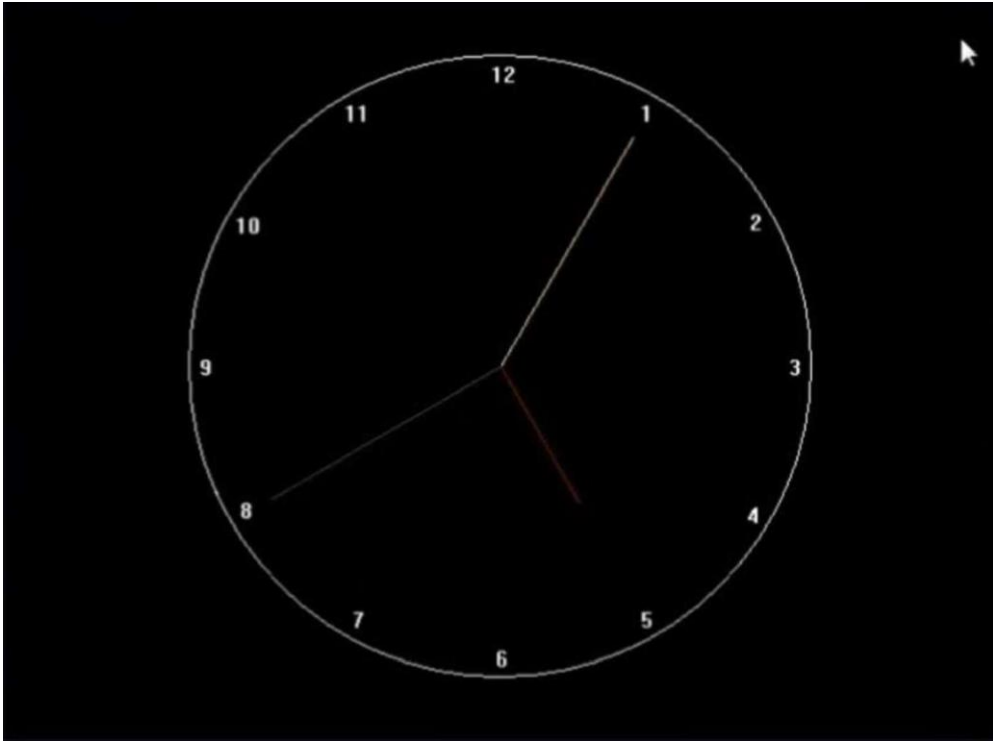
```

clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoryBlocks Settings Help
- global -
get_time(int &h, int &m, int &s): void
112 {
113 time_t rawtime;
114 struct tm *t;
115 time(&rawtime);
116 t = gmtime(&rawtime);
117 h=t->tm_hour;
118 m=t->tm_min;
119 s=t->tm_sec;
120 }
121 //=====
122 /** Function to draw Second needle. */
123 void second_needle(int s)
124 {
125 float angle=-90;
126 float sx,sy;
127 setcolor(0);
128 sx=cx+M_N_L*cos((angle+s*6)*M_PI/180);
129 sy=cy+M_N_L*sin((angle+s*6)*M_PI/180);
130 line(cx,cy,sx,sy);
131 setcolor(0);
132 sx=cx+M_N_L*cos((angle+s*6)*M_PI/180);
133 sy=cy+M_N_L*sin((angle+s*6)*M_PI/180);
134 line(cx,cy,sx,sy);
135 }
136 /** Function to draw Minute needle. */
137 void minute_needle(int m, int s)
138 {
139 float angle=-90;
140 float sx,sy;
141 setcolor(0);
142 sx=cx+M_N_L*cos((angle+m*6)*M_PI/180);
143 sy=cy+M_N_L*sin((angle+m*6)*M_PI/180);
144 line(cx,cy,sx,sy);
145 setcolor(0);
146 sx=cx+M_N_L*cos((angle+m*6+(s*6/60))*M_PI/180);
147 sy=cy+M_N_L*sin((angle+m*6+(s*6/60))*M_PI/180);
148 line(cx,cy,sx,sy);
149 }
150 }

```

```
clock.cpp - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoryBlocks Settings Help
- global -
Management
  Projects Files FSys
  Workspace
clock.cpp x
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.

REFERENCES

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>