PART- A

## Design the following using library functions (line, fillellipse, sefillstyle, floodfill, circle, rectangle, fillpoly, etc.)

#include<conio.h>

#include<graphics.h>

#include<iostream.h>

#include<dos.h>

#include<stdlib.h>

void main()

{

int gd,gm,i;

int arr[]={100,200,300,100,500,200,100,200} ;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\\turboc3\\bgi"); 7

setfillstyle(SOLID\_FILL,MAGENTA);

rectangle(150,200,435,400);

floodfill(250,300,WHITE);

setfillstyle(HATCH\_FILL,BROWN);

fillpoly(4,arr);

floodfill(300,150,WHITE);

line(280,200,280,400);

line(335,200,335,400);

line(280,300,335,300);

circle(307,300,10);

setfillstyle(XHATCH\_FILL,RED);

fillellipse(375,300,20,30);

floodfill(375,300,WHITE) ;

line(265,400,225,450);

line(350,400,315,450);

line(225,450,315,450);

setcolor(GREEN);

rectangle(180,250,275,315);

for(i=-7;i<=80;i+=16)

{

line(195+i,250,195+i,315);

}

line(180,280,275,280);

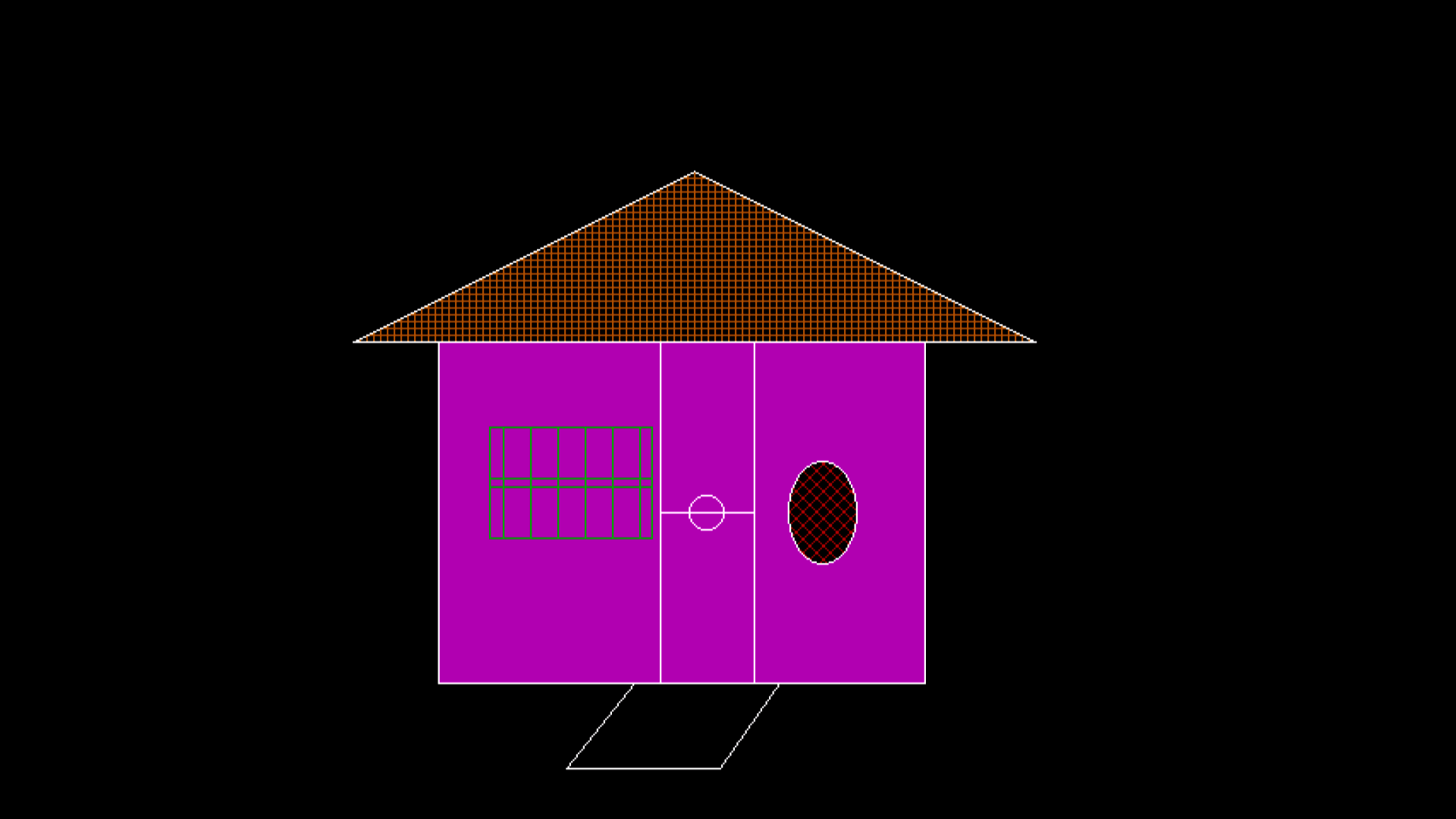
line(180,285,275,285);

getch();

closegraph();

}

Output:



## Design the following using library functions (line, fillellipse, sefillstyle, floodfill, circle, rectangle, fillpoly, pieslice, sector, arc, etc.)

#include<graphics.h>

#include<conio.h>

int main()

{

int gd,gm;

clrscr();

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

//This is to draw man's head

circle(100,200,25);

setfillstyle(SOLID\_FILL,YELLOW);

floodfill(101,201,15);

//This is to draw man's eyes

fillellipse(90,193,4,7);

setfillstyle(SOLID\_FILL,BLACK);

floodfill(90,194,15);

fillellipse(110,193,4,7);

setfillstyle(SOLID\_FILL,BLACK);

floodfill(111,194,15);

//This is to draw man's nose and mouth

setcolor(BLACK);

line(100,195,100,205);

pieslice(100,220,0,180,10);

//This is to draw man's legs and hands

setcolor(GREEN);

line(100,225,100,350);

line(70,230,100,245);

line(100,245,135,230);

line(100,350,70,370);

line(100,350,130,370);

//This is to draw birds

setcolor(WHITE);

arc(378,40,0,90,20);

arc(418,40,80,180,20);

arc(262,85,10,90,20);

arc(295,80,80,170,20);

// This is for pound rectangular box

int arr[]={200,350,600,350,600,475,200,475,200,350};

setfillstyle(SLASH\_FILL,BLUE);

fillpoly(5,arr);

//This is for fish body

setcolor(8);

ellipse(260,400,0,360,35,12);

setfillstyle(SOLID\_FILL,CYAN);

floodfill(280,400,8);

//This is for fish body arch and eye

setcolor(WHITE);

ellipse(240,400,315,45,15,15);

setcolor(RED);

circle(235,400,2);

//This is to draw fish wing

setfillstyle(SOLID\_FILL,GREEN);

setcolor(WHITE);

pieslice(280,393,60,190,14);

pieslice(280,405,190,320,14);

// This is to draw fish tale

setfillstyle(SOLID\_FILL,RED);

setcolor(BLACK);

sector(310,400,60,300,20,8);

//This is for stick

setcolor(WHITE);

line(135,230,250,388);

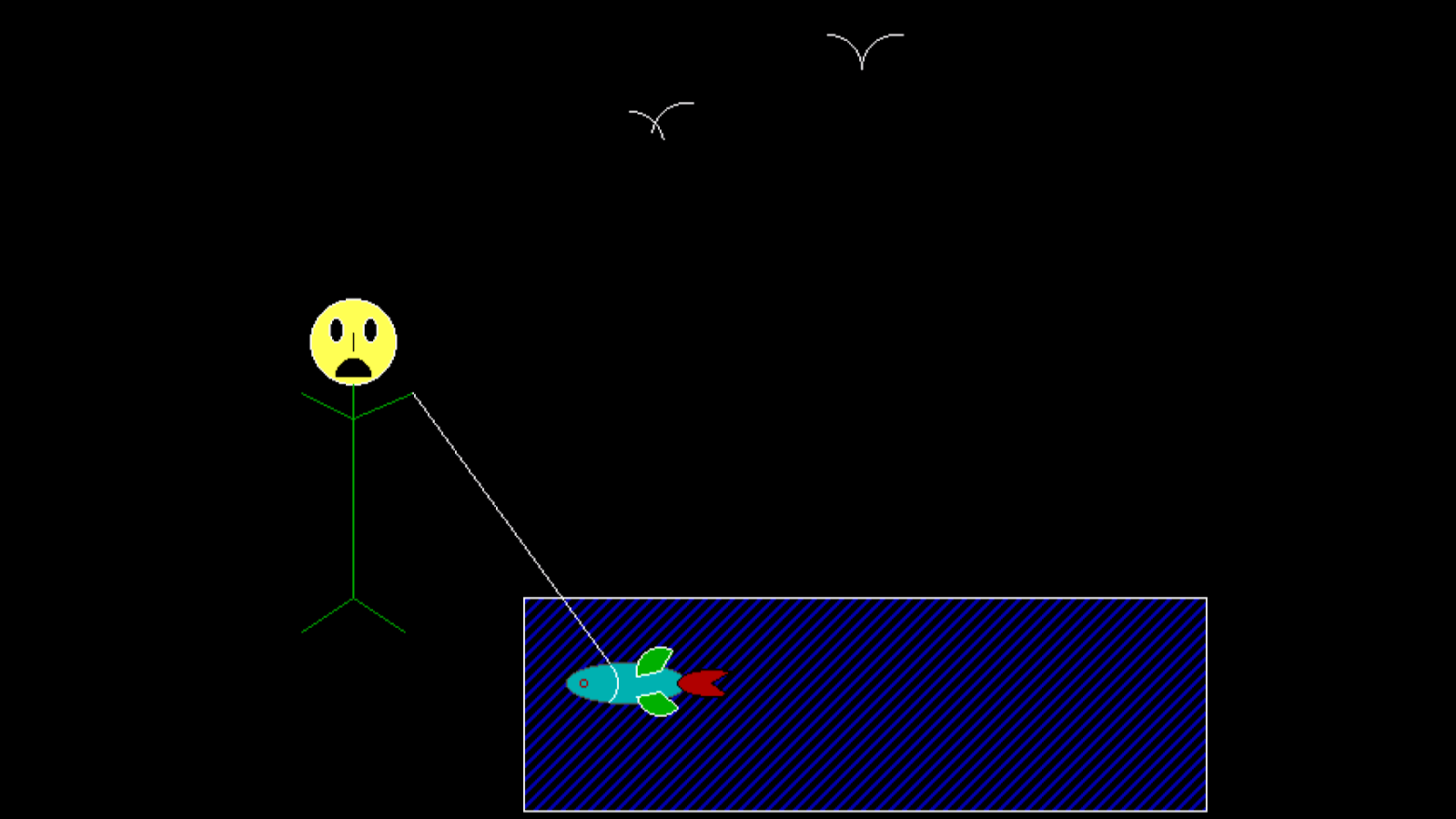
getch();

closegraph();

return 0;

}

Output:



## Program to implement moving car (design car using drawpoly, setfillstyle, floodfill, circle, line, etc.).

#include<graphics.h>

#include<conio.h>

#include<dos.h>

void main()

{

int gd,gm,i=0;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

for(i;i<420;++i)

{

int j[]={55+i,155,90+i,155,90+i,172,55+i,172,55+i,155};

int k[]={120+i,155,155+i,155,155+i,172,120+i,172,120+i,155};

setcolor(WHITE);

setfillstyle(BKSLASH\_FILL,MAGENTA);

circle(65+i,230,10);

floodfill(65+i,230,WHITE);

circle(145+i,230,10);

floodfill(145+i,230,WHITE) ;

setcolor(BLUE);

setfillstyle(SOLID\_FILL,WHITE);

drawpoly(5,j);

floodfill(80+i,160,BLUE);

drawpoly(5,k);

floodfill(130+i,160,BLUE);

line(0+i,180,30+i,180);

line(180+i,180,210+i,180);

line(30+i,180,70+i,125);

line(70+i,125,140+i,125);

line(140+i,125,180+i,180);

line(0+i,180,0+i,230);

line(210+i,180,210+i,230);

line(0+i,230,54+i,230);

line(76+i,230,134+i,230);

line(210+i,230,156+i,230);

delay(30);

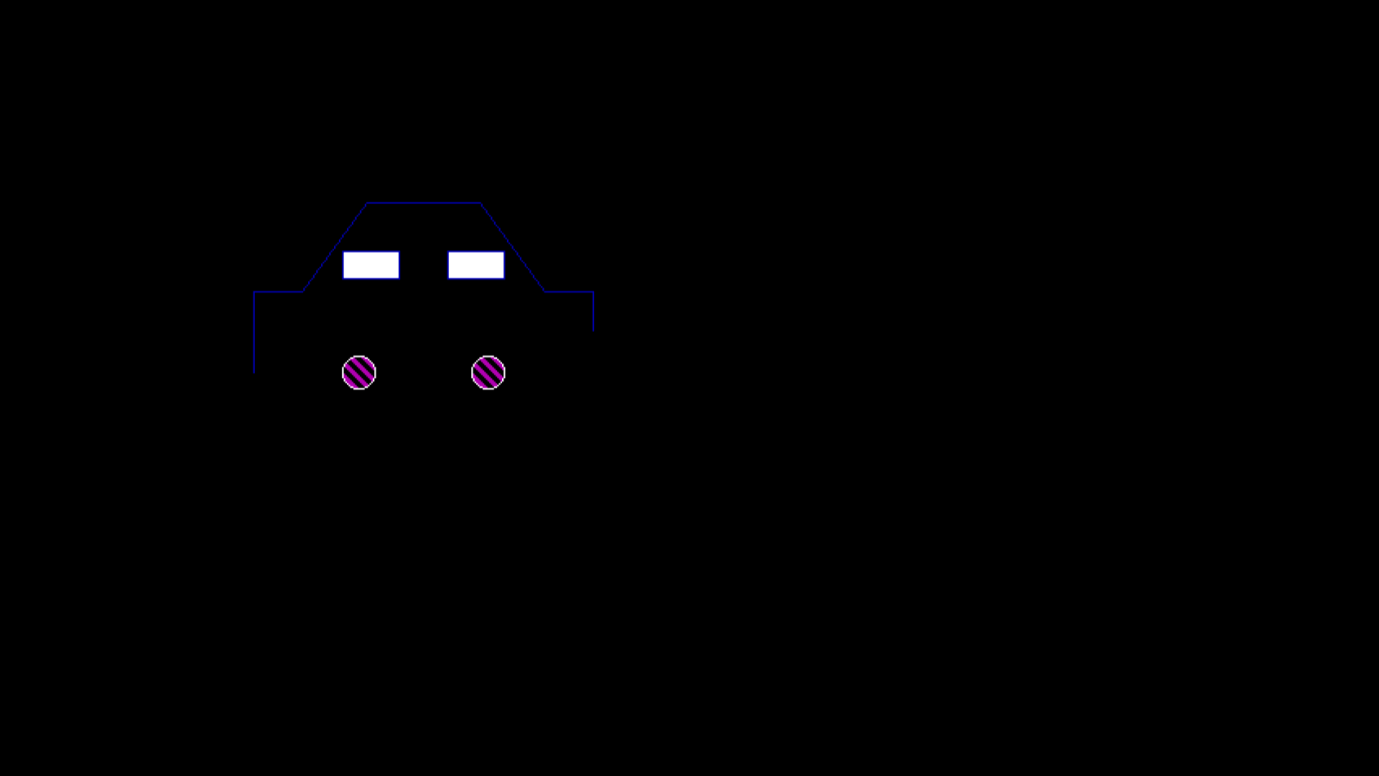
cleardevice();

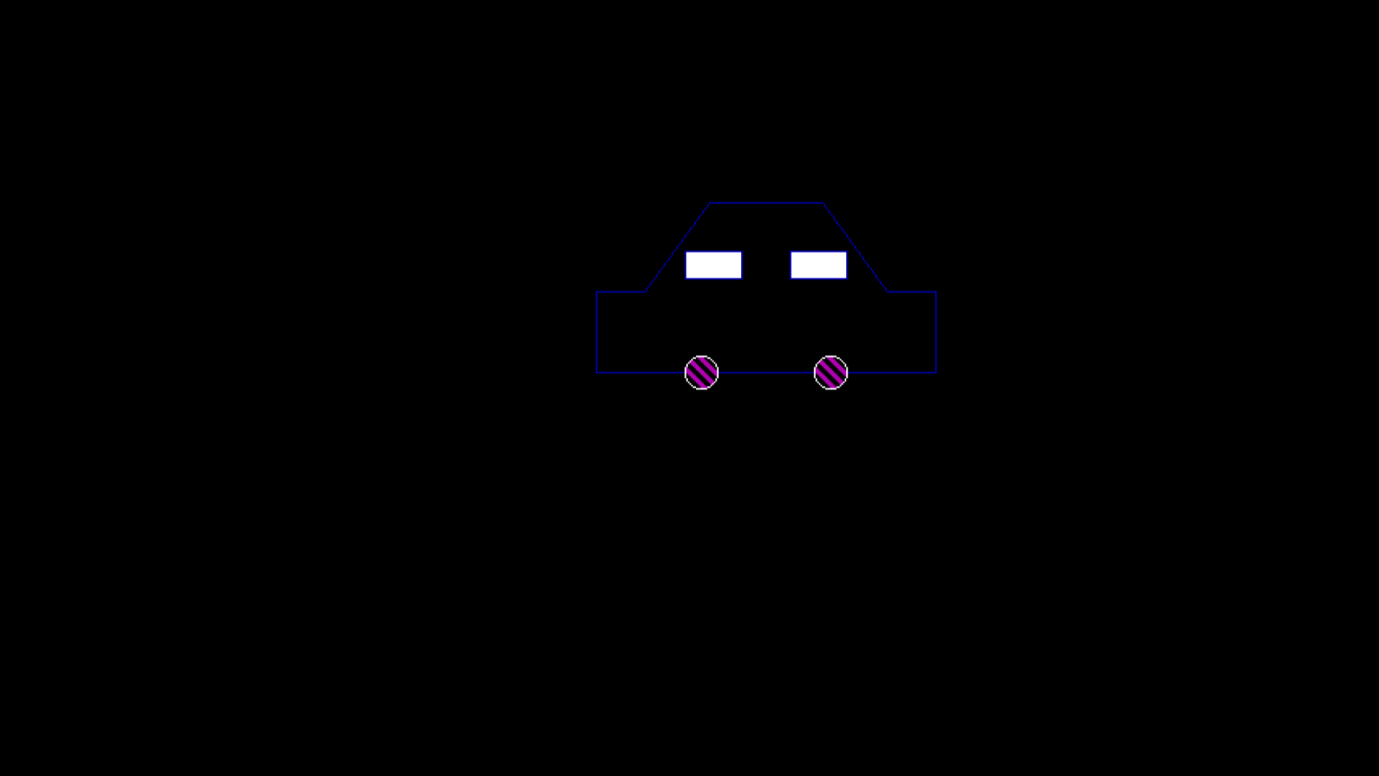
}

closegraph();

}

Output:





## Program to implement face animation with blinking eyes (circle, floodfill, fillellipse, line, etc.).

#include<graphics.h>

#include<conio.h>

#include<stdlib.h>

#include<dos.h>

void drawface(){

setcolor(YELLOW);

circle(50, 100, 25);

setfillstyle(SOLID\_FILL, YELLOW);

floodfill(50, 100, YELLOW);

}

void normal(){

setcolor(BLACK);

setfillstyle(SOLID\_FILL, BLACK);

fillellipse(44, 85, 2, 6);

fillellipse(56, 85, 2, 6);

ellipse(50, 100, 205, 335, 20, 9);

ellipse(50, 100, 205, 335, 20, 10);

ellipse(50, 100, 205, 335, 20, 11);

delay(1000);

}

void smile(){

setcolor(BLACK);

setfillstyle(SOLID\_FILL, BLACK);

setlinestyle(SOLID\_LINE,1,BLACK);

line(40, 85, 46, 85);

line(52, 85, 58, 85);

ellipse(50, 100, 0, -180, 20, 9);

ellipse(50, 100, 0, -180, 20, 10);

ellipse(50, 100, 0, -180, 20, 11);

delay(1000);

}

void drawText(){

setcolor(WHITE);

settextstyle(SANS\_SERIF\_FONT, HORIZ\_DIR, 2);

outtextxy(155, 451, "Smiling Face Animation");

}

void main()

{

int gd,gm;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

cleardevice();

while(!kbhit()){

drawface();

drawText();

normal();

cleardevice();

drawface();

drawText();

smile();

cleardevice();

}

getch();

closegraph();

}

Output:





## Program to implement digital clock to display current time (hh:mm:ss) with ticking sound.

// Program to implement digital clock

#include<graphics.h>

#include<time.h>

#include<string.h>

#include<conio.h>

#include<dos.h>

#include<iostream.h>

int main()

{

int gd,gm,midx,midy;

time\_t rawtime;

tm\* timeinfo;

char DClock[]="Digital Clock",HHMMSS [20];

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

// Mid pixel in horizontal and vertical axis

midx=getmaxx()/2;

midy=getmaxy()/2;

// Set color, justify and textstyle

setcolor(GREEN);

settextjustify(CENTER\_TEXT, CENTER\_TEXT);

settextstyle(4,HORIZ\_DIR,6);

// While loop to display the time till any key hit

while(!kbhit())

{

cleardevice();

//To display rectangle and Title

rectangle(midx -120,midy -40,midx +120,midy+ 40);

rectangle(midx -140,midy -60,midx +140,midy+ 60);

moveto(midx,350);

outtext(DClock);

// To get current time in HH:MM:SS format

time(&rawtime);

timeinfo=localtime(&rawtime);

strftime(HHMMSS,20,"%H:%M:%S",timeinfo);

// print current time at center

moveto(midx,midy);

outtext(HHMMSS);

// To give ticking sound and delay of 1 second

sound(400);

delay(30);

nosound();

delay(970);

}

closegraph();

return 0;

}

Output:

