BY-Prajakta Maam.

**name:-kishor.d.aldar**

**rollno:-6481**

**syit-d**

**computer graphics.**

**INTRODUCTION**

Using functions of graphics.h in Turbo C++ compiler you can make graphics programs, animations, projects, and games.We can draw circles, lines, rectangles, bars and many other geometrical figures.

In this code we have use graphics.h as a main folder.. actually our bgi files are not supporting…so we have copied all the files from bgi and pasted in bin.. so we do not required to mention our path in our code.

Then we have added timer code for our starting count down..10,9,8…0.

Then in the same function then we have continued our rocket game.in that we have used many header files.

PRESS enter when count down stops. Then game screen will appear press enter to start your rocket then some keys are used to move up, down left, right using that we have to collect the coins or points. Then you will get score on the right corner with your name then just be cool and play the game and your highscore will be saved in the files. For further players to target.

Outputs will help you to understand the code very neatly and ofcourse we have given code downward.

**CODE:-**

#include <graphics.h>

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include <dos.h>

#include <iostream.h>

#include <fstream.h>

#include <string.h>

void main()

{ {int i,midx,midy,gd=DETECT,gm,ec;

char a[10];

initgraph(&gd,&gm,"");

ec=graphresult();

if(ec|=0)

{

cout<<"error in graphics drvier";

getch();

exit(0);

}

settextjustify(CENTER\_TEXT,CENTER\_TEXT);

settextstyle(3,HORIZ\_DIR,25);

setcolor(RED);

midx=getmaxx()/2;

midy=getmaxy()/2;

for(i=10;i>=0;i--)

{

cleardevice();

sprintf(a,"%d",i);

circle(midx,midy,100);

circle(midx,midy,99);

outtextxy(midx+10,midy-20,a);

delay(1000);

cleardevice();

}

getch();

closegraph();

}

int gdriver = DETECT, gmode, errorcode;

void

\*body,\*food,\*tail1,\*tail2,\*tail3,\*tail4,\*head1,\*head2,\*head3,\*head4;

int x, y,X[5000],Y[5000],i=3,

maxx,maxy,speed=100,bo=10,t[10],score=0,hscore=20;

unsigned int size;

char a='6',b,scor[4],hs[4];

int k=2,l,r1,r2,f=0,z=100,first=0,second=1;;

r1=300;

r2=350;

ifstream infile("c:\tc\bin\rattle.txt");

infile.getline(hs,4);

infile.close();

hscore = atoi(hs);

/\* initialize graphics and local variables \*/

initgraph(&gdriver, &gmode, " ");

/\* read result of initialization \*/

errorcode = graphresult();

if (errorcode != grOk) /\* an error occurred \*/

{

printf("Graphics error: %s", grapherrormsg(errorcode));

printf("Press any key to halt:");

getch();

exit(1); /\* terminate with an error code \*/

}

maxx = getmaxx();

maxy = getmaxy();

y = 160;

x = 80;

for(int j=0; j<=3; j++)

{

X[i-j]=x-bo\*(j+1);

Y[i-j]=y;

}

//body

setfillstyle(1,2);

bar(10,10,20,20);

setcolor(0);

setlinestyle(0,1,3);

line(9,9,21,21);

line(9,21,21,9);

size = imagesize(10,10,20,20);

body = malloc(size);

getimage(10,10,20,20,body);

cleardevice();

setcolor(0);

//tail1-right

t[0]=20; t[1]=10;

t[2]=10; t[3]=10;

t[4]=20; t[5]=15;

t[6]=10; t[7]=20;

t[8]=20; t[9]=21;

fillpoly(5, t);

size = imagesize(10,10,20,21);

tail1 = malloc(size);

getimage(10,10,20,21,tail1);

cleardevice();

//tail2-left

t[0]=10; t[1]=10;

t[2]=20; t[3]=10;

t[4]=10; t[5]=15;

t[6]=20; t[7]=20;

t[8]=10; t[9]=21;

fillpoly(5, t);

size = imagesize(10,10,20,21);

tail2 = malloc(size);

getimage(10,10,20,21,tail2);

cleardevice();

//tail3-up

t[0]=10; t[1]=20;

t[2]=10; t[3]=10;

t[4]=15; t[5]=20;

t[6]=20; t[7]=10;

t[8]=20; t[9]=21;

fillpoly(5, t);

size = imagesize(10,10,20,21);

tail3 = malloc(size);

getimage(10,10,20,21,tail3);

cleardevice();

//tail4-down

t[0]=10; t[1]=10;

t[2]=10; t[3]=21;

t[4]=15; t[5]=10;

t[6]=20; t[7]=21;

t[8]=20; t[9]=10;

fillpoly(5, t);

size = imagesize(10,10,20,21);

tail4 = malloc(size);

getimage(10,10,20,21,tail4);

cleardevice();

//head1-right

setlinestyle(1,1,1);

setcolor(2);

fillellipse(10,10,10,5);

setcolor(4);

//eyes

fillellipse(15,7,1,1);

fillellipse(15,13,1,1);

size = imagesize(10,5,20,15);

head1 = malloc(size);

getimage(10,5,20,15,head1);

//head2-left

//eyes

fillellipse(5,7,1,1);

fillellipse(5,13,1,1);

size = imagesize(0,5,10,15);

head2 = malloc(size);

getimage(0,5,10,15,head2);

cleardevice();

//head3-up

//eyes

setcolor(2);

fillellipse(12,12,5,10);

setcolor(4);

fillellipse(9,6,1,1);

fillellipse(15,6,1,1);

size = imagesize(7,2,18,12);

head3 = malloc(size);

getimage(7,2,18,12,head3);

//head4-down

//eyes

fillellipse(9,17,1,1);

fillellipse(15,17,1,1);

size = imagesize(7,12,18,22);

head4 = malloc(size);

getimage(7,12,18,22,head4);

cleardevice();

//food

setcolor(15);

setfillstyle(1,15);

fillellipse(10,10,3,5);

size = imagesize(5,3,15,17);

food = malloc(size);

getimage(5,3,15,17,food);

cleardevice();

/\* repeat until a key is pressed \*/

maxx=getmaxx();

maxy=getmaxy();

setlinestyle(0,1,3);

setcolor(9);

rectangle(0,0,maxx,maxy);

setlinestyle(0,1,2);

line(maxx-150,0,maxx-150,maxy);

setcolor(8);

setfillstyle(1,8);

bar(maxx-147,3,maxx-3,63);

setcolor(12);

int maze=1;

if(maze==1)

{

setfillstyle(1,6);

bar(4,4,maxx-154,14);

bar(4,4,14,maxy-4);

bar(4,maxy-4,maxx-154,maxy-14);

bar(maxx-154-10,4,maxx-154,maxy-4);

}

settextstyle(0,0,1);

delay(300);

outtextxy(maxx-125,30,"RATTLE SNAKE");

delay(300);

setcolor(11);

setlinestyle(0,1,2);

rectangle(maxx-145,65,maxx-5,205);

outtextxy(maxx-128,70,"INSTRUCTIONS");

delay(200);

setcolor(6);

outtextxy(maxx-140,90,"Commands Key");

delay(300);

setcolor(8);

outtextxy(maxx-140,110," Move Up 8");

delay(50);

outtextxy(maxx-140,130," Move Down 5");

delay(50);

outtextxy(maxx-140,150," Move Right 6");

delay(50);

outtextxy(maxx-140,170," Move Left 4");

delay(50);

outtextxy(maxx-140,190," Exit 0");

delay(200);

setcolor(12);

setlinestyle(0,1,2);

rectangle(maxx-145,210,maxx-5,350);

setcolor(1);

outtextxy(maxx-130,220,"Player's Name");

setcolor(15);

outtextxy(maxx-100,230,"KISHOR");

setcolor(4);

delay(200);

itoa(score,scor,10);

outtextxy(maxx-140,250,"Score");

outtextxy(maxx-50,250,scor);

delay(200);

setcolor(10);

outtextxy(maxx-140,280,"Level");

delay(200);

setcolor(13);

outtextxy(maxx-140,310,"Maze 1");

delay(200);

setcolor(14);

outtextxy(maxx-140,340,"High Score ");

outtextxy(maxx-50,340,hs);

setcolor(4);

outtextxy(maxx-130,440,"Made By :- ");

setcolor(2);

outtextxy(maxx-110,460,"Kishor Aldar");

for(j=0; j<=1000; j++)

{

putpixel(random(maxx-150),random(maxy),6);

delay(1);

}

while (a!='0')

{

// plot new image

if(a=='6')

putimage(x, y, head1, XOR\_PUT);

if(a=='4')

putimage(x, y, head2, XOR\_PUT);

if(a=='8')

putimage(x, y, head3, XOR\_PUT);

if(a=='5')

putimage(x, y, head4, XOR\_PUT);

for(j=0; j<k; j++)

{

putimage(X[i-j], Y[i-j], body, XOR\_PUT);

}

if((X[i-j-1]-X[i-j])<0 && (Y[i-j-1]-Y[i-j])==0)

putimage(X[i-j], Y[i-j], tail1, XOR\_PUT);

if((X[i-j-1]-X[i-j])>0 && (Y[i-j-1]-Y[i-j])==0)

putimage(X[i-j], Y[i-j], tail2, XOR\_PUT);

if((X[i-j-1]-X[i-j])==0 && (Y[i-j-1]-Y[i-j])<0)

putimage(X[i-j], Y[i-j], tail3, XOR\_PUT);

if((X[i-j-1]-X[i-j])==0 && (Y[i-j-1]-Y[i-j])>0)

putimage(X[i-j], Y[i-j], tail4, XOR\_PUT);

delay(speed);

if(second==1)

{

setcolor(15);

outtextxy(maxx-140,360,"Press Any Key...");

getch();

setcolor(0);

outtextxy(maxx-140,360,"Press Any Key...");

a='6';

second=2;

}

if(first==1)

{

setcolor(15);

outtextxy(maxx-140,360,"Congratulations ");

sound(100);

delay(300);

nosound();

getch();

setcolor(0);

outtextxy(maxx-140,360,"Congratulations ");

second++;

first=2;

}

// erase old image

if(a=='6')

putimage(x, y, head1, XOR\_PUT);

if(a=='4')

putimage(x, y, head2, XOR\_PUT);

if(a=='8')

putimage(x, y, head3, XOR\_PUT);

if(a=='5')

putimage(x, y, head4, XOR\_PUT);

for(j=0; j<k; j++)

{

putimage(X[i-j], Y[i-j], body, XOR\_PUT);

}

if((X[i-j-1]-X[i-j])<0 && (Y[i-j-1]-Y[i-j])==0)

putimage(X[i-j], Y[i-j], tail1, XOR\_PUT);

if((X[i-j-1]-X[i-j])>0 && (Y[i-j-1]-Y[i-j])==0)

putimage(X[i-j], Y[i-j], tail2, XOR\_PUT);

if((X[i-j-1]-X[i-j])==0 && (Y[i-j-1]-Y[i-j])<0)

putimage(X[i-j], Y[i-j], tail3, XOR\_PUT);

if((X[i-j-1]-X[i-j])==0 && (Y[i-j-1]-Y[i-j])>0)

putimage(X[i-j], Y[i-j], tail4, XOR\_PUT);

if(f==0)

{

putimage(r1,r2,food,XOR\_PUT);

f=1;

}

z--;

if((x>=r1 && y>=r2 && x<=r1+10 && y<=r2+10) || (x<=r1 && y<=r2 &&

x>=r1-10 && y>=r2-10) || z==0)

{

if(z!=0)

{

sound(800);

delay(20);

setcolor(0);

score += 10;

outtextxy(maxx-50,250,scor);

itoa(score,scor,10);

if(score>hscore)

{

strcpy(hs,scor);

setfillstyle(1,1);

bar(maxx-140,337,maxx-20,348);

setcolor(14);

outtextxy(maxx-50,340,hs);

outtextxy(maxx-140,340,"High Score ");

if(first==0)

{

first=1;

}

}

setcolor(4);

outtextxy(maxx-50,250,scor);

k++;

nosound();

}

z=100;

putimage(r1,r2,food,XOR\_PUT);

repr1:

r1=random(450);

if(r1<50)

goto repr1;

repr2:

r2=random(400);

if(r2<50)

goto repr2;

f=0;

}

i++;

X[i]=x;

Y[i]=y;

b=a;

if(kbhit())

{

rep:

a=getche();

if((b=='6' && a=='4') || (b=='4' && a=='6'))

a=b;

if((b=='8' && a=='5') || (b=='5' && a=='8'))

a=b;

if(a!='6' && a!='4' && a!='8' && a!='0' && a!='5')

a=b;

}

if(a=='8')

y -= bo;

if(a=='5')

y += bo;

if(a=='4')

x -= bo;

if(a=='6')

x += bo;

for(j=i+1; j<i-k; j--)

{

if((Y[i]>=Y[j] && Y[i]<=Y[j]+10) || (Y[i]<=Y[j] && Y[i]>=X[j]-10))

{

if((X[i]>=X[j] && X[i]<=X[j]+10) || (X[i]<=X[j] && X[i]>=X[j]-10))

{

a='0';

}

outtextxy(maxx-140,360,"C ");

}

if((X[i]>=X[j] && X[i]<=X[j]+10) || (X[i]<=X[j] && X[i]>=X[j]-10))

{

if((Y[i]>=Y[j] && Y[i]<=Y[j]+10) || (Y[i]<=Y[j] && Y[i]>=X[j]-10))

{

a='0';

}

outtextxy(maxx-140,360,"C ");

}

}

if(maze==1)

{

if(x<=14 || x>=maxx-184 || y<=14 || y>=maxy-14)

{

a='0';

}

}

}

if(score>hscore)

{

ofstream onfile("c:\tc\bin\rattle.txt");

onfile<<scor;

onfile.close();

}

setcolor(15);

outtextxy(maxx-140,390," Gameover");

sound(100);

delay(400);

nosound();

getch();

/\* clean up \*/

free(body);

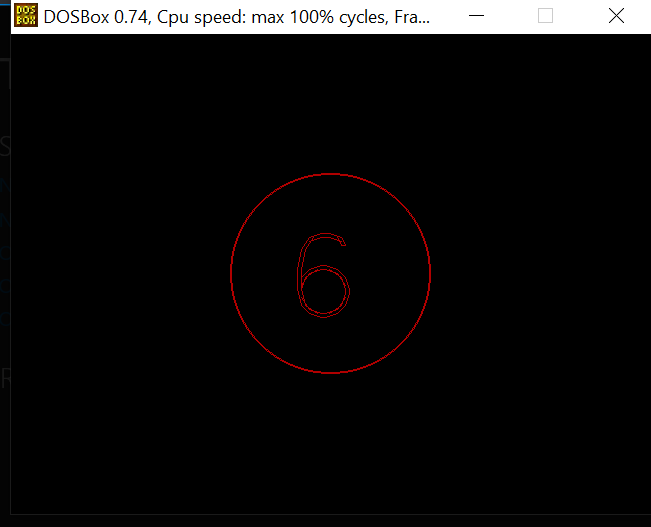
closegraph();

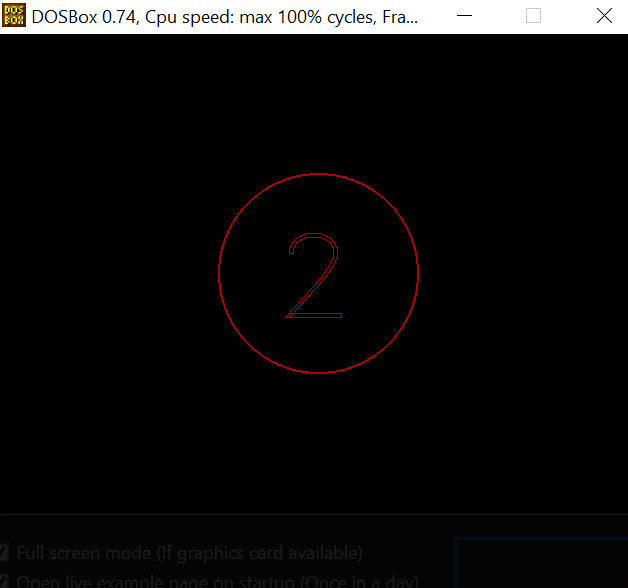
}

**OUTPUT:-**

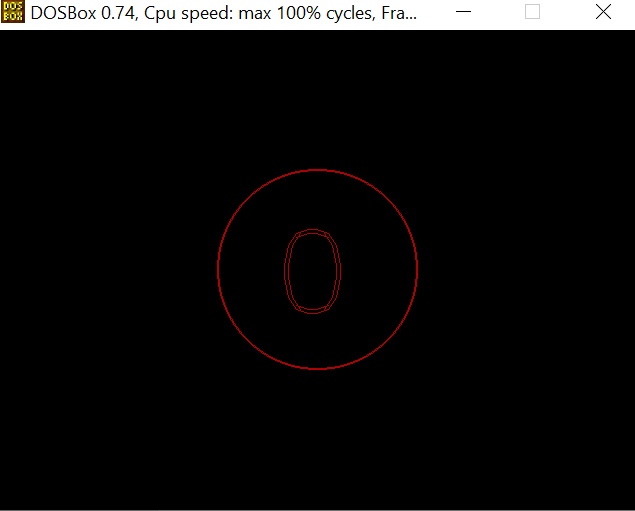
TIMER WILL START WHEN YOU START RUNNING THE CODE

10..9..8 LIKE THAT





TILL GET 0.. AFTER THAT OUR ROCKET GAME WILL BEGIN.



THIS IS OUR GAME ROCKET HAVE TO COLLECTS THAT POINTS ..PLAYER NAME KISHOR.





FINALLY WHNE GAMEOVER WHEN WE CRASH TO WALL THAT TIME IT WILL CREATE A SOUND AND END.

