MINI PROJECT REPORT

ON

"GUN-DART GAME USING C- GRAPHICS"

Submitted in fulfilment of the requirements

For the degree of

BACHELOR OF COMPUTER ENGINEERING

BY

GROUP MEMBER 1:-	**NAME:- KALYANI SINGH**	**ROLL NO:- SE-A-149**
GROUP MEMBER 2:-	**NAME:- KISHAN GUPTA**	**ROLL NO:- SE-A-111**

Under the guidance of
Prof. Shital K. Dhamal

LOKMANYA TILAK COLLEGE OF ENGINEERING
Department Of Computer Engineering

YEAR 2020-2021

CERTIFICATE

This is to certify that the mini project entitled **" GUN-DART GAME USING C GRAPHICS"** is a bonafide work of **'KALYANI SINGH' Roll no- (SE-A-149)** and **'KISHAN GUPTA' Roll no- (SE-A-111) .** It is submitted to the University of Mumbai in fulfillment of the requirement for the award.

Prof. Shital K. Dhamal
Guide

ACKNOWLEDGEMENT

Here we gladly present this mini project report on **" GUN DART GAME USING C "** as a part of the 3rd semester BE in Computer Engineering. At this time of submitting this report we use this opportunity to mention those people who were with us for this work. We extend our sincere and heartfelt thanks to our esteemed **guide, Prof. Shital Dhamal** for providing us with the right guidance and advice at the crucial junctures and for showing us the right way.

KALYANI SINGH

(SE-A-149)

Name Of the Student

KISHAN GUPTA

(SE-A-111)

Name Of the Student

INDEX

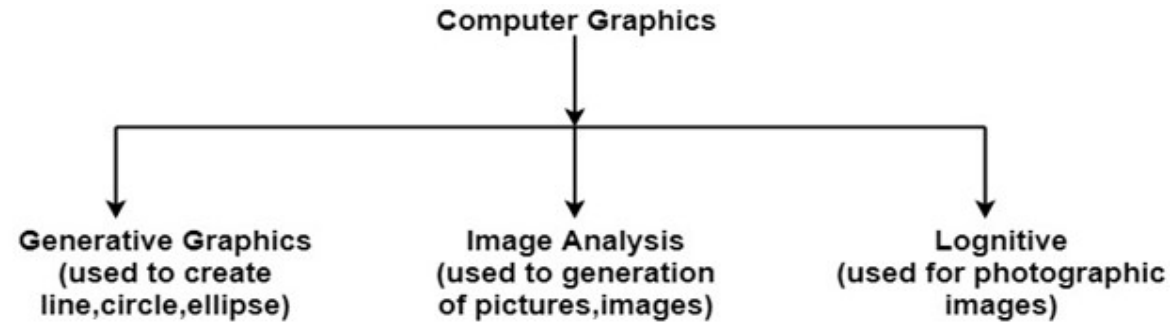
INTRODUCTION:-

- The game **"gun dart"** was build with the objective of understanding the c graphics library in a much more practical approach.

- Concepts such as frame buffer, 3D tralation, scaling, rotation have been utilized to understand the core concept of image viewing and processing.

- Concepts about the graphics library was understood better with the variety of functions which are in-built in the library and come handy in carrying out not only some of the most minor tasks but also major graphics processes like rendering, shading, sprite graphics, rendering, ray tracing, geometry processing, computer animation, vector graphics, 3D modeling, shaders, GPU design, implicit surface visualization, image processing, computational

photography,scientific visualization, computational geometry and computer vision, among others

- 

- Computer graphics is the branch of computer science that deals with generating images with the aid of computers.
- Today, computer graphics is a core technology in digital photography, film, video games, cell phone and computer displays, and many specialized applications.
- A great deal of specialized hardware and software has been developed, with the displays of most devices being driven by computer graphics hardware. It is a vast and recently developed area of computer science.
- The phrase was coined in 1960 by computer graphics researchers Verne Hudson and William Fetter of Boeing. It is often abbreviated as CG, or typically in the context of film as computer generated imagery (CGI).

- The overall methodology depends heavily on the underlying sciences of geometry, optics, physics, and perception.

- Computer graphics is responsible for displaying art and image data effectively and meaningfully to the consumer. It is also used for processing image data received from the physical world, such as photo and video content.

- Computer graphics development has had a significant impact on many types of media and has revolutionized animation, movies, advertising, video games, and graphic design in general.

Program:-

```
#include<stdio.h>

#include<graphics.h>

#include<conio.h>

#include<string.h>



int main() {
   int gm, gd = DETECT, radius;
   initgraph(& gd, & gm ,"C:\TURBOC3\BGI");

   int T = 0;
   int F = 0;
   char str[50], point[50], tPoint[50];
   boolean Start = false;
   int currentPoint = 0, totalPoint = 0, shotLeft = 10;
   int DownScroll = 0;

   // Main Loop
   int page=0;
   do {

      setactivepage(page);
      setvisualpage(1-page);
       cleardevice();
      int xBullet = 850 - F, yBullet = 362.5;



      if (Start == false) {
```

```
rectangle(0, 0, 1275, 700);
setfillstyle(1, 9);
floodfill(1, 1, 9);
bar3d(50,50,350,200,30,1);
setfillstyle(1,13);
floodfill(51,51,15);
settextstyle(8,0,4);
outtextxy(65,60,"GUN-DART GAME");
outtextxy(115,110,"Using C");
setfillstyle(1,5);
floodfill(60,49,15);
floodfill(351,190,15);
rectangle(70,325+50,340,575+50);
setfillstyle(1,0);
floodfill(71,326+50,15);

setfillstyle(1,10);
bar3d(50,230,350,350,20,1);
floodfill(51,231,15);
setfillstyle(1,2);
floodfill(60,229,15);
floodfill(352,300,15);
settextstyle(8,0,3);
outtextxy(53,233,"BY:");
                        outtextxy( 70,265, "KALYANI (SEA-149)");
// outtextxy(150,270,"& ");
outtextxy(70,300,"KISHAN (SEA-111)");



setfillstyle(1,15);
circle(200,450+50,10);
floodfill(201,451+50,15);

                        setfillstyle(1,4);
circle(200,450+50,40);
```

```
    floodfill(212,461+50,15);
                        setfillstyle(1,1);
                        circle(200,450+50,70);
                        floodfill(261,461+40+50,15);
                        setfillstyle(1,14);
                        circle(200,450+50,120);
    floodfill(271,521+50,15);

                        bar3d(400, 50, 1100, 600,60,1);
    setfillstyle(1, 11);
    floodfill(651, 51, 15);
    setfillstyle(1,3);
    floodfill(410,49,15);
    floodfill(1101,510,15);

    settextstyle(8, 0, 6);
    outtextxy(400, 100, "WELCOME TO THE GAME!!!");
    settextstyle(8,0,4);
                        outtextxy(500, 240, "You Have 10 Shots Per Game ");

    outtextxy(450, 300, " points for Yellow Disk --> 50");
    outtextxy(450, 350, " points for Blue Disk    --> 75");
    outtextxy(450, 400, " points for Red Disk     --> 100");

    outtextxy(400, 500, "Press Any Key To Start The Game...");




                        Start = true;
    getch();

}

// Title
```

```
    outtextxy(900, 5, "Press Esc Key To Exit The Game");
    outtextxy(50, 5, "Press Left Arrow Key To Fire The Bullet");
    outtextxy((1300-DownScroll), 675, "This Game is Created By Kishan Gupta and Kalyani Singh Under The Guidance of Miss. Shital Dhamal. Year - 2020");
    settextstyle(3, 0, 5);

    // Boundaries
    rectangle(0, 0, 20, 700); //left boundary
    rectangle(1245, 0, 1280, 700);//right boundary
    rectangle(0, 0, 1280, 30);//top boundary
    rectangle(0, 670, 1280, 700);//bottom  boundary
                line(75,30,75,670);// line for the target to not mix with the boundary
                setfillstyle(WIDE_DOT_FILL,15);
                floodfill(74,31,15);

    // Filling Corner

    setfillstyle(1, 12);
    floodfill(1, 1, 15); // Top Left Corner
    floodfill(1, 672, 15); // Bottom Left Corner
    floodfill(1279, 1, 15); // Top Right Corner
    floodfill(1279, 672, 15); // Bottom Right Corner




    //creating background lanscape
    int ar[]={400,150,575,35,750,150,400,150}; //roof poly co-ordinates
                setfillstyle(1,1);
                rectangle(400,150,750,370);
                floodfill(401,151,15);
                setfillstyle(1,4);
          drawpoly(4,ar); //roof
    floodfill(570,145,15);
                setfillstyle(1,12);
```

```
int pr[]={516,370,516,250,634,250,634,370,516,370}; //door poly co-ordinates
drawpoly(5,pr);
floodfill(517,369,15);



    // left tree


            int tr[]={150,395,150,300,180,300,180,395,150,395};
setfillstyle(1,6);
drawpoly(5,tr);
floodfill(151,394,15);
            int br[]={80,300,130,250,200,250,250,300,80,300};
            setfillstyle(1,10);
            drawpoly(5,br);
            floodfill(110,299,15);
            int fr[]={90,250,140,200,190,200,240,250,90,250};
            drawpoly(5,fr);
            floodfill(110,249,15);
            int zr[]={100,200,165,150,230,200,100,200};
            drawpoly(4,zr);
            floodfill(110,199,15);



//left top tree


    int cr[]={150+150,395-100,150+150,300-100,180+150,300-100,180+150,395-100,150+150,395-100};
setfillstyle(1,6);
drawpoly(5,cr);
floodfill(151+150,394-100,15);
            int nr[]={80+150,300-100,130+150,250-100,200+150,250-100,250+150,300-100,80+150,300-100};
            setfillstyle(1,2);
            drawpoly(5,nr);
            floodfill(110+150,299-100,15);
```

```
                int xr[]={90+150,250-100,140+150,200-100,190+150,200-100,240+150,250-100,90+150,250-100};
                drawpoly(5,xr);
                floodfill(110+150,249-100,15);
                int qr[]={100+150,200-100,165+150,150-100,230+150,200-100,100+150,200-100};
                drawpoly(4,qr);
                floodfill(110+150,199-100,15);



  //right tree


                int     mr[]={150+700,395-100,150+700,300-100,180+700,300-100,180+700,395-100,150+700,395-
100};
    setfillstyle(1,6);
     drawpoly(5,mr);
     floodfill(151+700,394-100,15);
                int dr[]={80+700,300-100,130+700,250-100,200+700,250-100,250+700,300-100,80+700,300-100};
                setfillstyle(1,2);
                drawpoly(5,dr);
                floodfill(110+700,299-100,15);
                int er[]={90+700,250-100,140+700,200-100,190+700,200-100,240+700,250-100,90+700,250-100};
                drawpoly(5,er);
                floodfill(110+700,249-100,15);
                int rrr[]={100+700,200-100,165+700,150-100,230+700,200-100,100+700,200-100};
                drawpoly(4,rrr);
                floodfill(110+700,199-100,15);



//right tree extreme


                int                 end[]={150+700+300,400-100-20,150+700+300,350-100-20,180+700+300,350-100-
20,180+700+300,400-100-20,150+700+300,400-100-20};
    setfillstyle(1,6);
     drawpoly(5,end);
     floodfill(151+700+300,399-100-20,15);
```

```
                int endd[]={80+700+300,350-100-20,130+700+300,300-100-20,200+700+300,300-100-
20,250+700+300,350-100-20,80+700+300,350-100-20};
                setfillstyle(1,10);
                drawpoly(5,endd);
                floodfill(110+700+300,349-100-20,15);
                int endr[]={90+700+300,300-100-20,140+700+300,250-100-20,190+700+300,250-100-
20,240+700+300,300-100-20,90+700+300,300-100-20};
                drawpoly(5,endr);
                floodfill(110+700+300,299-100-20,15);
                int endrp[]={100+700+300,250-100-20,165+700+300,200-100-20,230+700+300,250-100-
20,100+700+300,250-100-20};
                drawpoly(4,endrp);
                floodfill(110+700+300,249-100-20,15);



      //creating base and man with the gun



                // line(950,670,950,405); // tester line for the man
                // line(950,405,1245,405); // tester line


       //shoes
         setfillstyle(1,6);
         rectangle(975,650-110,1025,665-110);    //left shoe
         rectangle(1005,665-110,1025,668-110);


          rectangle(975+90,650-110,1025+90,665-110); //right shoe
         rectangle(1005+90,665-110,1025+90,668-110);
      floodfill(977,651-110,15);
      floodfill(1067,651-110,15);
      setfillstyle(1,8);
      floodfill(1006,666-110,15);
      floodfill(1101,666-110,15);


    //legs
      setfillstyle(1,1);
```

```
            int             legs[]={975+17,650-110,975+35,540-110,975+100+15,540-110,1125-17,650-110,1125-40,650-
110,975+75,570-110,975+40,650-110,975+17,650-110};
                    drawpoly(7,legs);
      floodfill(975+18,649-110,15);



   //belt
      setfillstyle(7,6);
            int belt[]={975+35,540-110,975+35,530-110,975+100+15,530-110,975+100+15,540-110,975+35,540-110};
                    drawpoly(5,belt);
                      floodfill(975+36,539-110,15);
                      setfillstyle(1,15);
                      circle(975+35+40,535-110,5);
              floodfill(1051,534-110,15);







           //hand sleeve
          setfillstyle(1,3);
          rectangle(975+15,365,975+55,390);
             floodfill(975+16,366,15);

   //hand
   setfillstyle(1,14);
          rectangle(975-50,370,975+15,385);
                                floodfill(975-49,371,15);



          //gun
          setfillstyle(1,8);
          int gun[]={925,370-25,925,420-25,910,420-25,910,400-25,860,400-25,860,370-25,925,370-25};
          drawpoly(7,gun);
          floodfill(924,346,15);
          arc(910,400-25,190,270,10);
```

```
        arc(911,400-24,190,270,10);
        arc(912,400-23,190,270,10);
        arc(911,400-25,190,270,10);
        arc(912,400-25,190,270,10);

        //shirt
                setfillstyle(1,9);
    int shirt[]={975+35,530-110,975+35,470-110,975+100+15,470-110,975+100+15,530-110,975+35,530-110};
     drawpoly(5,shirt);
    floodfill(975+36,529-110,15);

  //neck
            setfillstyle(1,14);
            int neck[]={975+35+30,470-110,975+35+30,460-110,975+35+50,460-110,975+35+50,470-
110,975+35+30,470-110};
            drawpoly(5,neck);
           floodfill(975+35+31,469-110,15);
          //eyes
          setfillstyle(1,0);
            circle(975+56,413-110,6);
            floodfill(975+56,413-110,15);

  //head
  setfillstyle(12,14);
            arc(975+75,460-40-110,146,34,40);
                line(975+40,460-63-110,975+75+35,460-63-110);
```

```
                        floodfill(975+75,460-40-110,15);
                       //arc(975+75,460-40,330,110,50);
                       line(975+10,397-110,975+75+35+30,397-110);
        line(975+10,396-110,975+75+35+30,396-110);
                        line(975+10,395-110,975+75+35+30,395-110);
                        line(975+10,394-110,975+75+35+30,394-110);
//          arc(975+10+65,397,0,180,50);

//cap



setfillstyle(1,6);
int cap[]={975+10+25,394-110,975+10+25,355-110,975+10+105,355-110,975+10+105,394-110,975+25,394-110};
drawpoly(5,cap);
          floodfill(975+10+26,393-110,15);




          //setfillstyle(1,3);
                    //  setcolor(6);
          //          line(75,370,1245,370);
          //          floodfill(21+55,349,15);
                     rectangle(190,590,910,670);
          //          setfillstyle(1,9);
          //          floodfill(21+60,440,15);



           //creating the partion in the main page
      setfillstyle(1,11);
      line(74,410,1011,410);
      floodfill(79,409,15);
      setfillstyle(1,10);
      line(1090,410,1245,410);
      floodfill(79,415,15);
```

```
        // Creating Target

    rectangle(25, 25 + T, 65, 200 + T); // First Target Slide  : width-30 , height -175
    line(25, 65 + T, 65, 65 + T);
    setfillstyle(1, 5);
    floodfill(26, 64 + T, 15);

    line(25, 100 + T, 65, 100 + T);
    setfillstyle(1, 1);
    floodfill(26, 99 + T, 15);

    line(25, 125 + T, 65, 125 + T);
    setfillstyle(1, 8);
    floodfill(26, 124 + T, 15);

    line(25, 160 + T, 65, 160 + T);
    setfillstyle(1, 1);
    floodfill(26, 159 + T, 15);

    line(25, 200 + T, 65, 200 + T);
    setfillstyle(1, 5);
    floodfill(26, 199 + T, 15);
                // ---------------------------------------This is For Bottom to Top----------------------------------------

//      rectangle(25,525-T,65,700-T);
//
//      line(25, 565 - T, 65, 565 - T);
//      setfillstyle(1, 14);
//      floodfill(26, 564 - T, 15);
```

```
//
//      line(25, 600 - T, 65, 600 - T);
//      setfillstyle(1, 9);
//      floodfill(26, 599 - T, 15);
//
//
//      line(25, 625 - T, 65, 625 - T);
//      setfillstyle(1, 4);
//      floodfill(26, 624 - T, 15);
//
//      line(25, 660 - T, 65, 660 - T);
//      setfillstyle(1, 9);
//      floodfill(26, 659 - T, 15);
//
//      line(25, 700 - T, 65, 700 - T);
//      setfillstyle(1, 14);
//      floodfill(26, 699 - T, 15);



      // IMPLEMENTING HIT
       settextstyle(8,0,4);
      if (xBullet <= (50) && xBullet >= (20) && yBullet > (25 + T) && yBullet < (65 + T) ) {
         currentPoint = 50;
         totalPoint += currentPoint;
         // textcolor(2);
          settextstyle(8,0,4);
         outtextxy(500, 100, "Good Shot");
         delay(1000);

      } else if (xBullet <= (50) && xBullet >= (20) && yBullet >= (65 + T) && yBullet < (100 + T)) {
         currentPoint = 75;
         totalPoint += currentPoint;
         //textcolor(3);
          settextstyle(8,0,4);
```

```
    outtextxy(480, 100, "Great Shot");
    delay(1000);



} else if (xBullet <= (50) && xBullet >= (20) && yBullet >= (100 + T) && yBullet < (125 + T)) {
    currentPoint = 100;
    totalPoint += currentPoint;
    //textcolor(4);
     settextstyle(8,0,4);
    outtextxy(450, 100, "Excellent Shot");
    delay(1000);

} else if (xBullet <= (50) && xBullet >= (20) && yBullet >= (125 + T) && yBullet < (160 + T)) {
    currentPoint = 75;
    totalPoint += currentPoint;
    //textcolor(3);
     settextstyle(8,0,4);
    outtextxy(480, 100, "Great Shot");
    delay(1000);

} else if (xBullet <= (50) && xBullet >= (20) && yBullet >= (160 + T) && yBullet <= (200 + T)) {
    currentPoint = 50;
    totalPoint += currentPoint;
    //textcolor(2);
     settextstyle(8,0,4);
    outtextxy(500, 100, "Good Shot");
    delay(1000);
}

// Creating Three boxes for Shot Left , Points , Total Points



settextstyle(3, 0, 1);

rectangle(200, 600, 400, 660); // Shots Left
```

```
        outtextxy(215, 601, "SHOTS LEFT");
        line(200, 625, 400, 625); // Line Separate
        sprintf(str, "%d", shotLeft);
        outtextxy(230, 626, str);

        rectangle(450, 600, 650, 660); // Points
        outtextxy(470, 601, "POINTS");
        line(450, 625, 650, 625);
        sprintf(point, "%d", currentPoint);
        outtextxy(491, 626, point);

        rectangle(700, 600, 900, 660); // Total Points
        outtextxy(715, 601, "TOTAL POINTS");
        line(700, 625, 900, 625);
        sprintf(tPoint, "%d", totalPoint);
        outtextxy(740, 626, tPoint);

        // Creating Bullet

       // int arr[] = {    1200,
       //     250,
         //   1200,
         //   265,
          //  1150,
          //  262.5,
          //  1200,
           // 250
       // };
         // This is Dummy Bullet
       // drawpoly(4, arr);
//          int gun[]={925,370-25,925,420-25,910,420-25,910,400-25,860,400-25,860,370-25,925,370-25};
        if (GetAsyncKeyState(VK_LEFT) || F > 0) { // This is For Firing the bullet

           int arr[] = {
               900 - F,
```

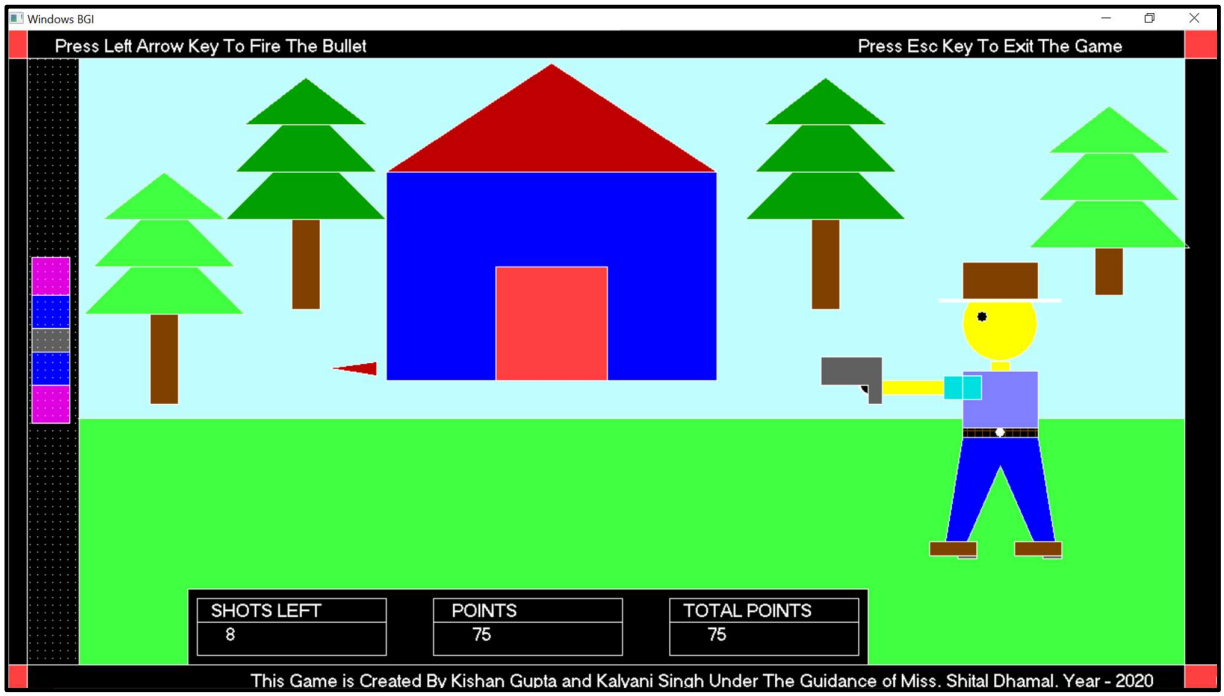
```
        350,
        900 - F,
        365,
        850 - F,
        357,
        900 - F,
        350
    };
    drawpoly(4, arr);
    setfillstyle(1,4);
    floodfill(895 - F, 360, 15);
    F = (F + 30) % 1200;              // Circular Increment of Bullet

    if (F == 840) {
        shotLeft--;
        if (xBullet <= (50) && xBullet >= (20) && yBullet <= (25 + T) || yBullet >= (200 + T)) {
            currentPoint = 0;
             settextstyle(8,0,4);
            outtextxy(500, 100, "Missed");
            delay(500);
        }
    }
}

// Increment Operation



T = (T + 5) % 500; // Circular increment of the Whole Screen
                DownScroll = (DownScroll + 1) % 1300;
delay(5);

if (GetAsyncKeyState(VK_ESCAPE)) { // For Any Time Exit
    break;
}
```

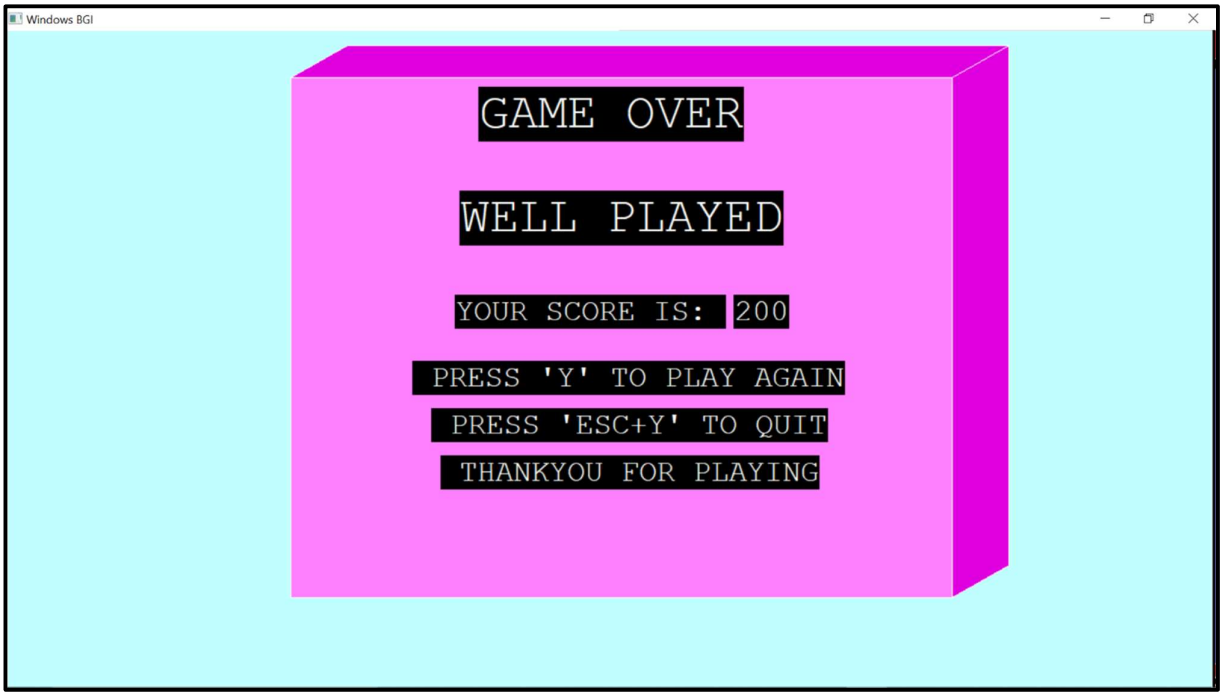
```
  if (shotLeft == 0) {
     delay(1000);
  }
 // Implementing Game Over PopUp
// Implementing Game Over PopUp
  if(shotLeft == 0){
  do {
     cleardevice();
      rectangle(0, 0, 1275, 700);
     setfillstyle(1, 11);
     floodfill(1, 1, 15);
     bar3d(300, 50, 1000, 600,60,1);
     setfillstyle(1, 13);
     floodfill(301, 51, 15);
     setfillstyle(1,5);
     floodfill(310,49,15);
     floodfill(1001,510,15);

     settextstyle(8, 0, 6);
     outtextxy(500, 60, "GAME OVER");
      outtextxy(480, 170, "WELL PLAYED");
     settextstyle(8,0,4);
                              outtextxy(475, 280, "YOUR SCORE IS: ");
                              outtextxy(770, 280, tPoint);
     outtextxy(430, 350, " PRESS 'Y' TO PLAY AGAIN");
     outtextxy(450, 400, " PRESS 'ESC+Y' TO QUIT");
     outtextxy(460, 450, " THANKYOU FOR PLAYING");
      /*  rectangle(550, 50, 1000, 550);
     setfillstyle(WIDE_DOT_FILL, 4);
     floodfill(651, 51, 15);

     settextstyle(3, 0, 3);
     outtextxy(710, 100, "Game Over");
     outtextxy(650, 180, "Your Score is :");
     outtextxy(1100, 180, tPoint);
```

```
        outtextxy(650, 280, "Press 'Y' To Play Again");
        outtextxy(575, 350, "Press 'Y' + 'Esc' To Exit The Game");
                                */
                    setactivepage(page);
                                setvisualpage(1-page);
                cleardevice();
                page = 1 - page;

    // Waiting For User's Input
        } while (!GetAsyncKeyState(0x59) && shotLeft == 0); // 0x59 == Y

        shotLeft = 10;
        currentPoint = 0;
        totalPoint = 0;
        F = 0;
        T = 0;
        DownScroll = 0;
    }
   page=1-page;

  } while (shotLeft <= 10);

  getch();
  closegraph();
  return 0;
}
```

OUTPUT:-


Windows BGI
GUN-DART GAME
Using C
BY:
KALYANI (SEA-149)
KISHAN (SEA-111)
WELCOME TO THE GAME!!!
You Have 10 Shots Per Game
points for Yellow Disk --> 50
points for Blue Disk --> 75
points for Red Disk --> 100
Press Any Key To Start The Game...


Windows BGI
Press Left Arrow Key To Fire The Bullet
Press Esc Key To Exit The Game
SHOTS LEFT
8
POINTS
75
TOTAL POINTS
75
This Game is Created By Kishan Gupta and Kalyani Singh Under The Guidance of Miss. Shital Dhamal. Year - 2020


Windows BGI
GAME OVER
WELL PLAYED
YOUR SCORE IS: 200
PRESS 'Y' TO PLAY AGAIN
PRESS 'ESC+Y' TO QUIT
THANKYOU FOR PLAYING

CONCLUSION:-

- The importance of computer graphics lies in its applications. In engineering applications (e.g. automotive and aerospace), the ability to quickly visualize newly designed shapes is indispensable.

- Before the advent of computer graphics, designers built expensive prototypes and time-consuming clay models.

- Now, designers can interactively view and modify models of their shapes using a computer.

- Medical imaging is another application where computer graphics has proven valuable.

- Recent advances in imaging technology such as computer tomography and magnetic resonance imaging allow physicians to take 3D Xrays of the human body.

- Interactive computer graphics allows the physician to interpret this large volume of data in new and useful ways.

- Computer graphics has also expanded the boundaries of art and entertainment.

- Movies such as Jurassiac Park make extensive use of computer graphics to create images that test the bounds of imagination.

- The development of computer graphics has made possible virtual reality, a synthetic reality that exists only inside a computer.

- Virtual reality is fast becoming an indispensable tool in education. Flight simulators are used to train pilot for extreme conditions.

- Surgical simulators are used to train novice surgeons without endangering patients.