# Archery Game In C


**BY:**

**Neha Justa 031281**

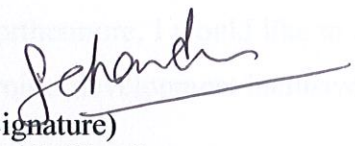



**Submitted in partial fulfillment of the Degree of Bachelor of Technology**


**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY-WAKNAGHAT**

# CERTIFICATE

This is to certify that the work entitled, **"Archery Game in C"** submitted by Neha Justa in partial fulfillment for the award of degree of Bachelor of Technology in 2007 of Jaypee University of Information Technology has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.



(Signature)
Satish Chandra
Sr. Lecturer
Department Of Computer Science & Engineering

## Acknowledgement

It is indeed a privilege as well as a pleasant duty to express my gratitude to all those who have made it possible for me to complete my project. I would like to express my deep sense of gratitude and heartiest thanks to my teacher ***Mr. Satish Chandra,*** Senior Lecturer, Jaypee University of Information Technology for guiding me throughout this project and providing me each and every resource required to make this project a success. I am sure the time I have spent with him will be a constant sourse of inspiration for me in the days to come.

Furthermore, I would like to thank all the friends who helped me directly or indirectly in project development inclusive of typical programming involved in this project.

Lastly, I convey my high gratitude and deep respect to the entire faculty for showing me different methodologies used in my project.

- Neha Justa 031281

# Table of Contents

# Introduction

**Archery Game**

This project has been made using C. This is a game designed using various features of C. The aim of the game is to shoot the balloons and score as much points.

The rules of the game are as following:-

Rule # 1 > Only One Player Can Play The Game At A time.
Rule # 2 > There Are Three Levels Of The game.
Rule # 3 > Arrow Shooter Is Positioned By The UP AND DOWN KEY.
Rule # 4 > Only One Arrow Can Be Shooted At A Time.
Rule # 5 > SpaceBar Is Used To Shoot The Arrow.
Rule # 6 > Enter Key Is Used To Select any Option.
Rule # 7 > Five(5) Points Is Rewarded To Each Hit One
Rule # 8 > One Can Go To Another Level ,Crossing The Previous Level.
Rule # 9 > Esc Key From The Main Will Terminate The Game.
Rule # 10 > Exit Option Will Terminate The Game.
Press Any Key To Resume The game...."

The Game comprises of three levels.The description of the various levels are as following:-

LEVEL # 1 > Four balloons will be moving vertically and the player has to shoot the balloon. Each hit is rewarded by 5 points. Once the score becomes 100, the player moves to another level.The player cannot skip or jump any level.

LEVEL # 2 > Five monsters will be moving horizontally, the player has to shoot the ball. Each hit is rewarded by 10 points. Once the score becomes 200, the player moves to another level. If the ball touches the shooter , particular game ends then and there.

LEVEL # 3 > One Eye will be enclosed in convex object moving vertically, the player has to target and shoot the eye.The hit which is closest to the eye will be rewarded accordingly. Once the player hits the eye exactly, the player is awarded with 500 points. Only ten arrows are provided for this level. If the player succeeds, the game comes to an end, else the player's game comes to an end.

# Playing the GAME

The first screen of the project consist of five menu options viz.

- Play the GAME
- Instructions
- High Score
- Level Description
- Exit

Any of the menu option can be selected either by using keyboard or left click of the mouse on the specified option.

Play the game,starts a new game

Instruction displays the rules of the game

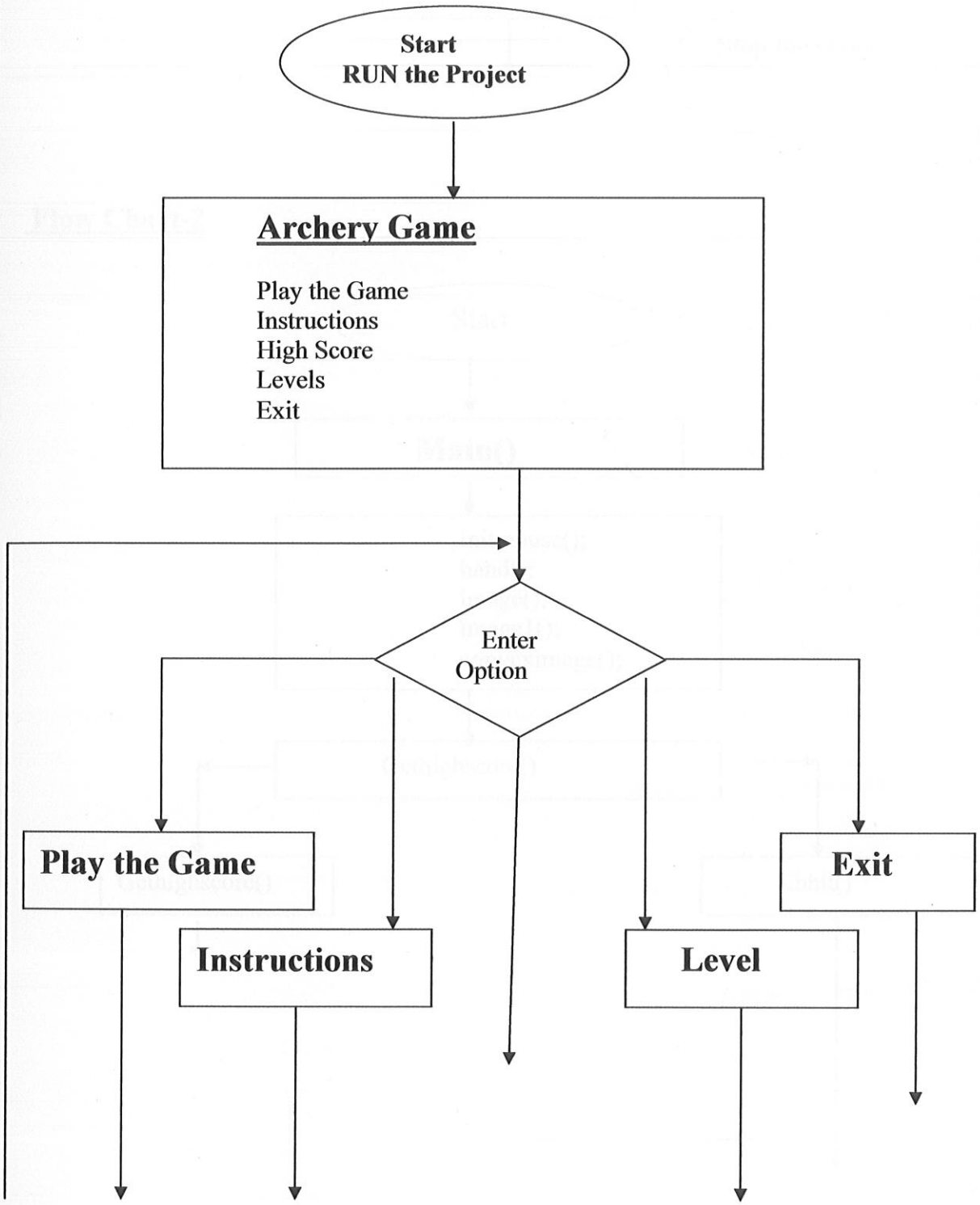
High Score displays the top five scorers and their respective scores

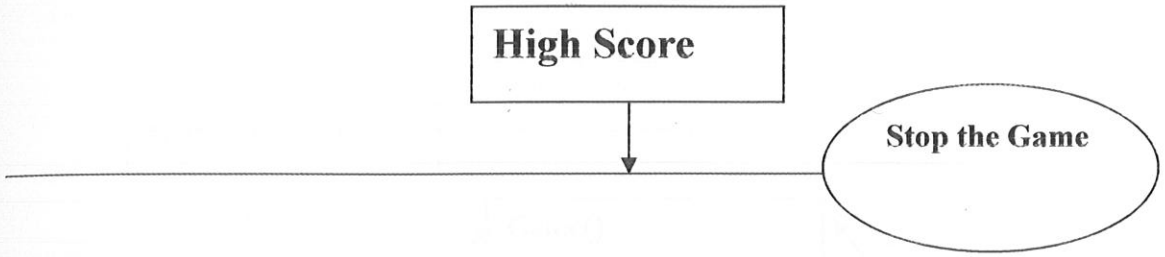
Level describes the three levels of the game\

Exit takes the control out of the game

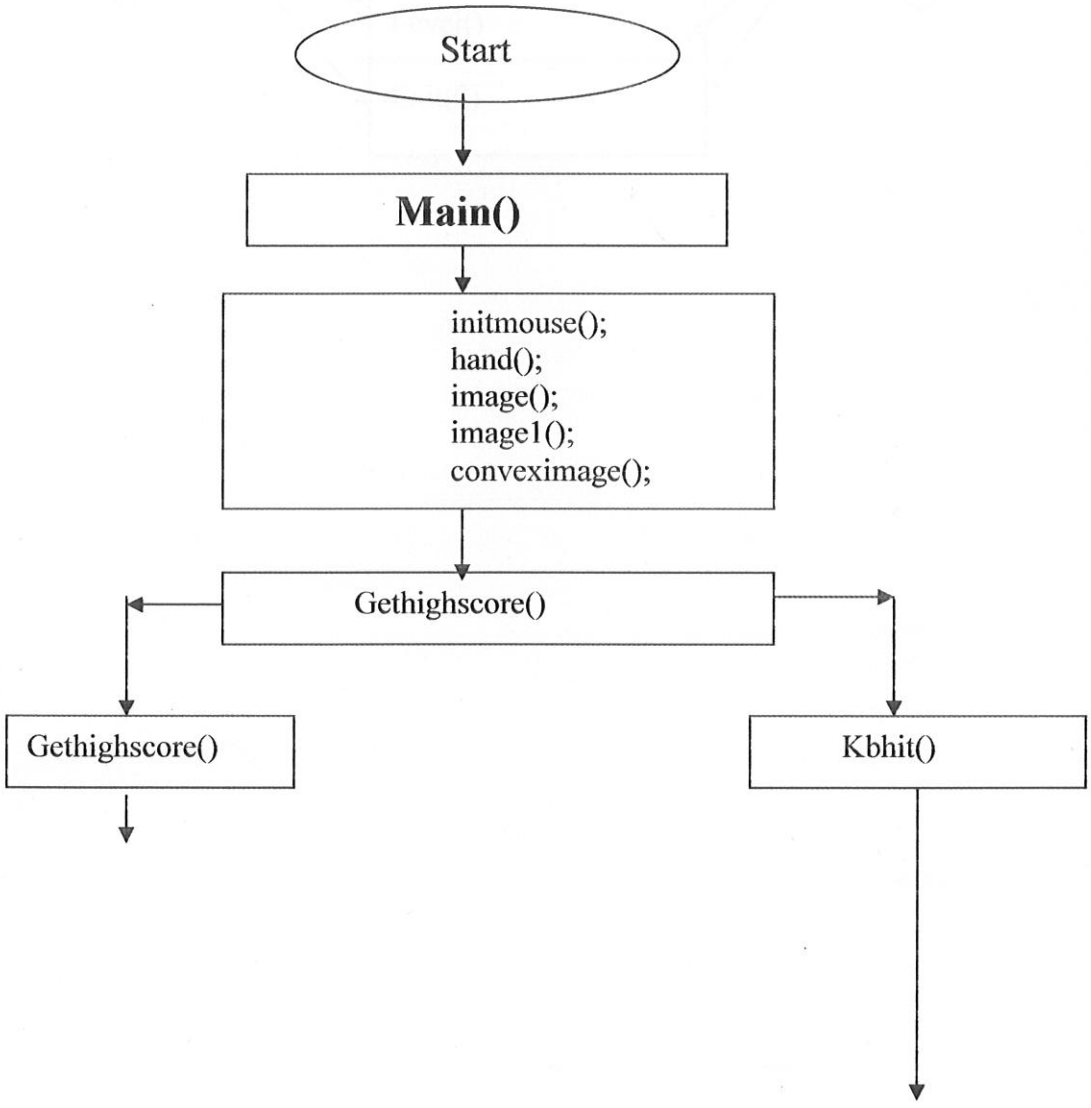
-

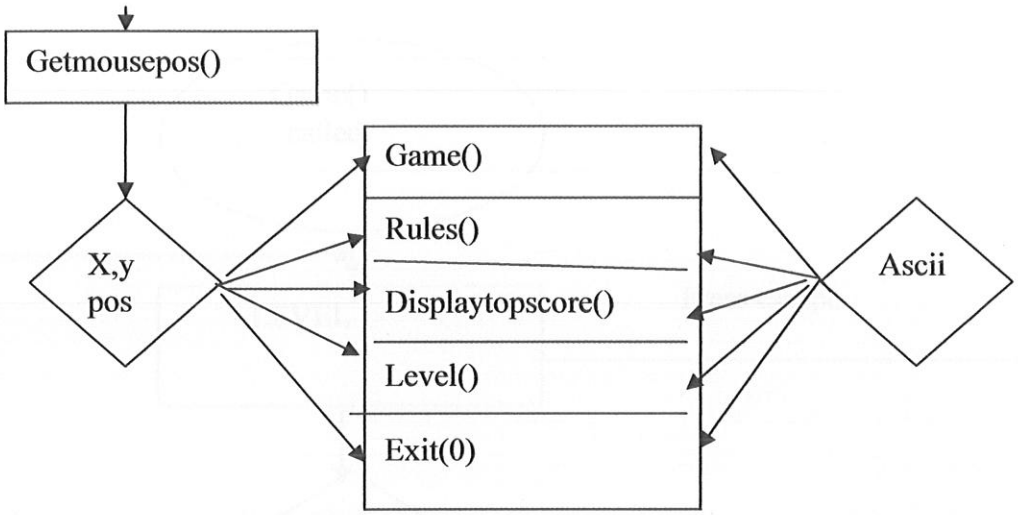
**Flow Chart-1**


Start
RUN the Project
Archery Game
Play the Game
Instructions
High Score
Levels
Exit
Enter
Option
Play the Game
Instructions
Level
Exit

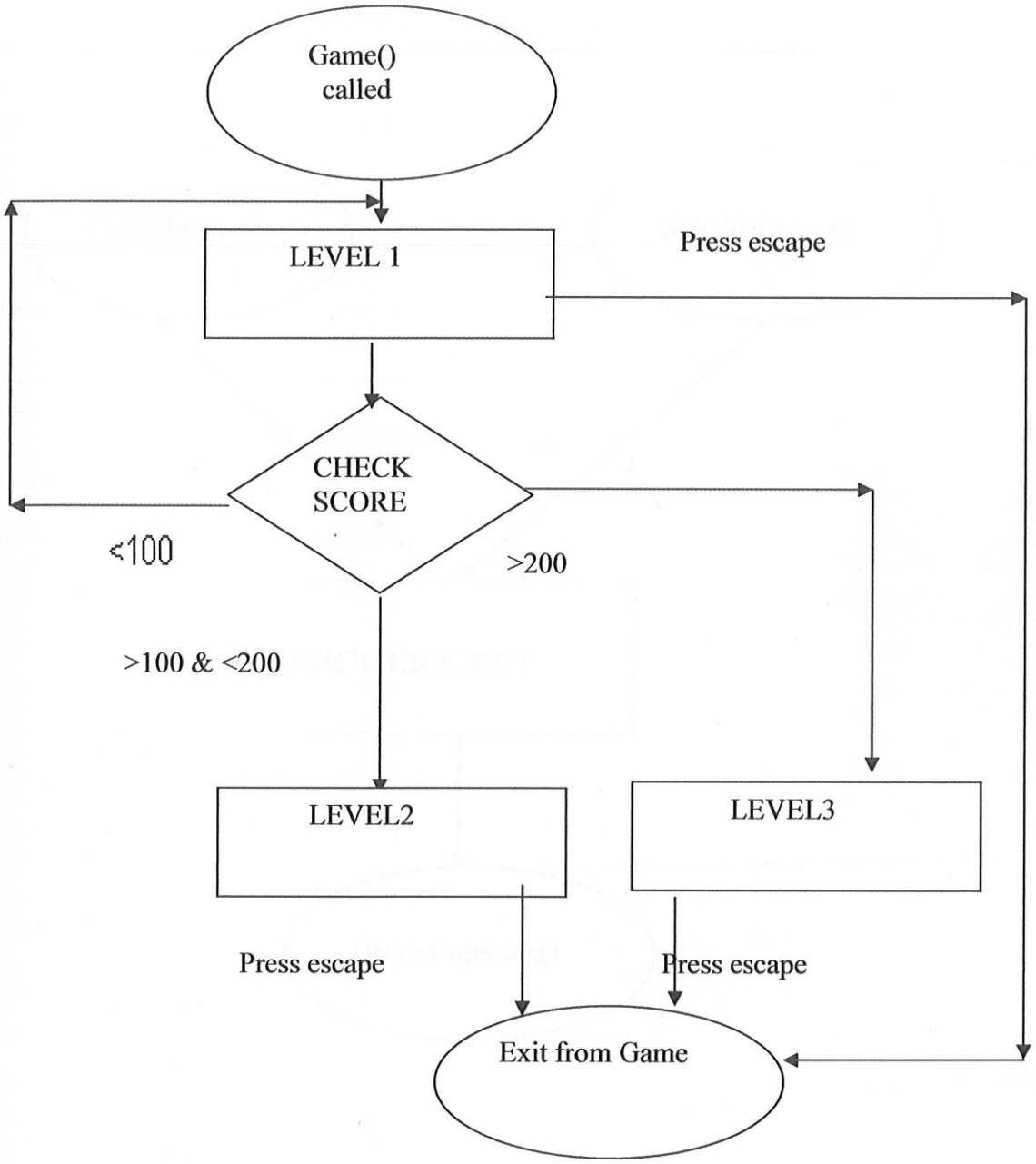

High Score
Stop the Game

**Flow Chart-2**

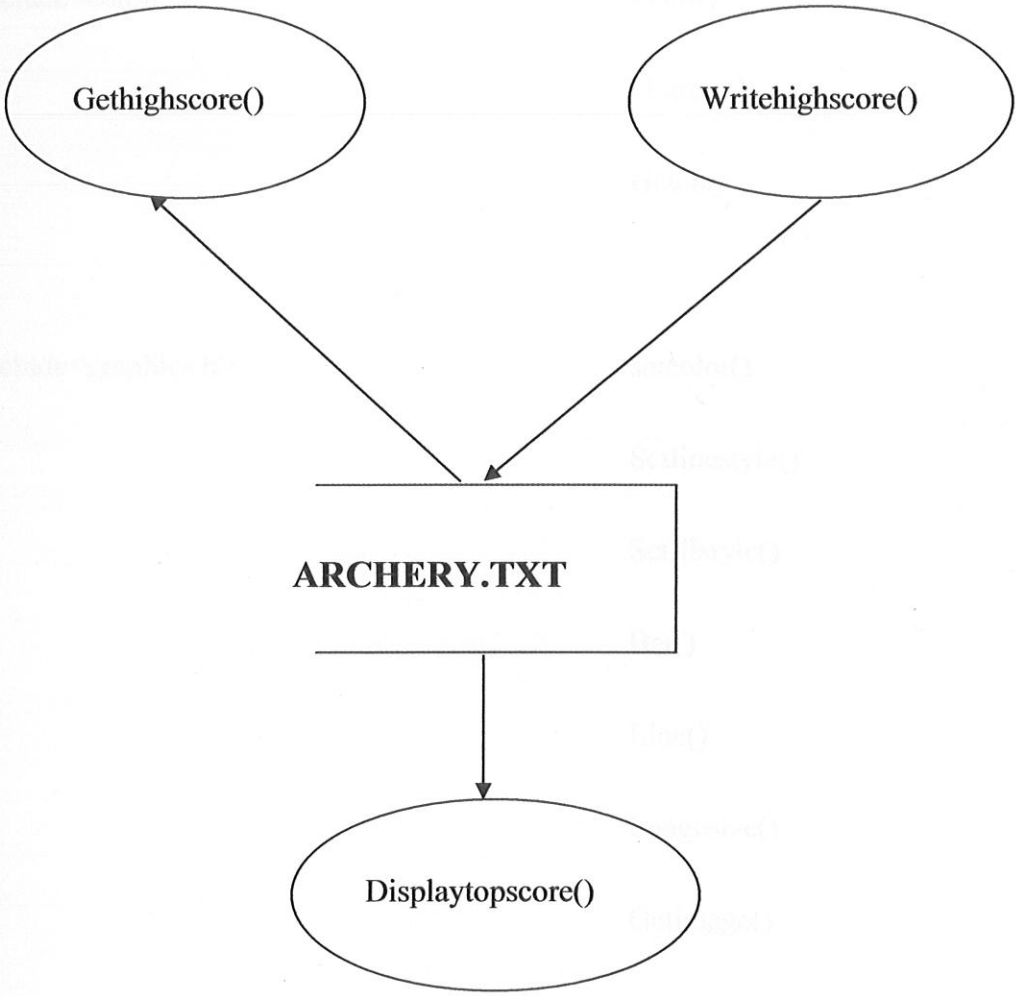

Start
Main()
initmouse();
hand();
image();
image1();
conveximage();
Gethighscore()
Gethighscore()
Kbhit()


Getmousepos()
X,y
pos
Game()
Rules()
Displaytopscore()
Level()
Exit(0)
Ascii

**Flow Chart -3**


Game()
called
LEVEL 1
Press escape
CHECK
SCORE
<100
>200
>100 & <200
LEVEL2
LEVEL3
Press escape
Press escape
Exit from Game

**Data Flow Diagram**



# List of header files & Inbuilt functions used

| HEADER FILES | INBUILT FUNCTIONS |
|---|---|
| #include<conio.h> | kbhit() |
| | Gotoxy() |
| | Getch() |
| #include<graphics.h> | setcolor() |
| | Setlinestyle() |
| | Setfilstyle() |
| | Bar() |
| | Line() |
| | Imagesize() |
| | Getimage() |
| | Cleardevice() |
| | Setcolor() |
| | Circle() |
| | Floodfill() |
| | Putimage() |

| | |
|---|---|
| | Ellipse() |
| | Putpixel() |
| | Fillellipse() |
| #include<string.h> | strcpy() |
| #include<process.h> | exit() |
| #include<dos.h> | REGS structure |
| | Sizeof() |
| | Delay() |
| | Int86() |
| #include<stdio.h> | sprintf() |
| | FILE |
| | Scanf() |
| | Fopen() |
| | Fprintf() |
| | Fclose() |

Printf()

EOF()

#include<stdlib>

malloc()

Itoa()

Random()

# Functions used in the project

- **void getkey();**

  Detecting the keyboard interrupt generated
- **void initmouse();**

  Initializes the mouse pointer
- **void showmouse();**

  Shows the mouse pointer on the screen
- **void hidemouse();**

  Hides the mouse pointer
- **void placemouse();**

  Places the cursor at a specific location
- **void getmousepos(int *,int *,int *);**

  Gets the xy –coordinates of the mouse pointer
- **void image1();**

  Image for level 1 (balls falling vertically)
- **void image();**

  Image for level 2 (balls sliding horizontally)
- **void conveximage();**

  Image for level 3 (eye)
- **void hand();**

  Function to print hand as a marker for menu
- **void arrow(int);**

  To draw the arrow ant the shooter
- **int shoot(int);**

  Coordinates of arrow and ball are matched
- **void game();**

  Initiates the various levels of the game
- **void ball();**

Moving balls in level 1

- **void eyelevel();**

  Moving eye in level 3

- **void balllevel2();**

  Moving balls in level 2

- **void gethighscores();**

  Geeting the score from the user

- **void displaytopscore();**

  Displaying the score through menu

- **void updatehighscore();**

  Display high score

- **void writehighscorestodisk();**

  Saving high score in a file

- **void rules();**

  Displaying instructions through menu

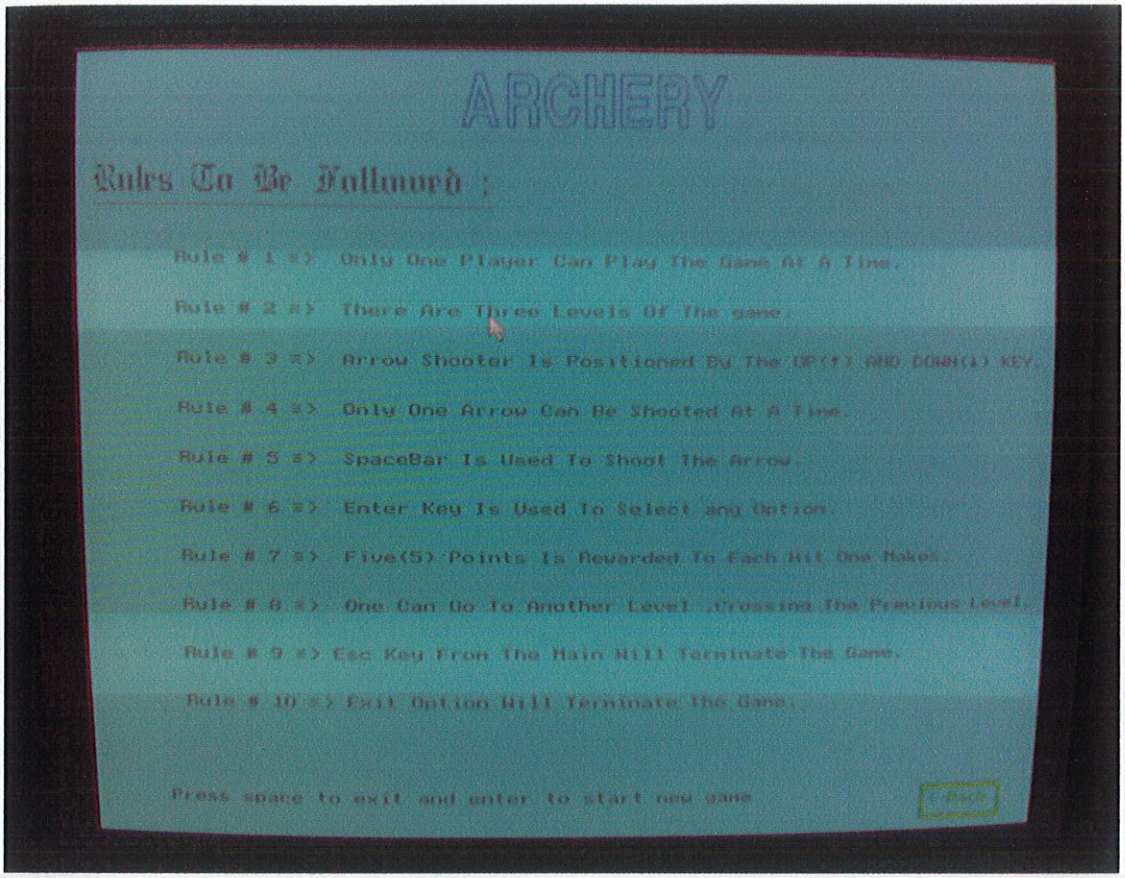
- **void level();**

  Description of various levels in the game

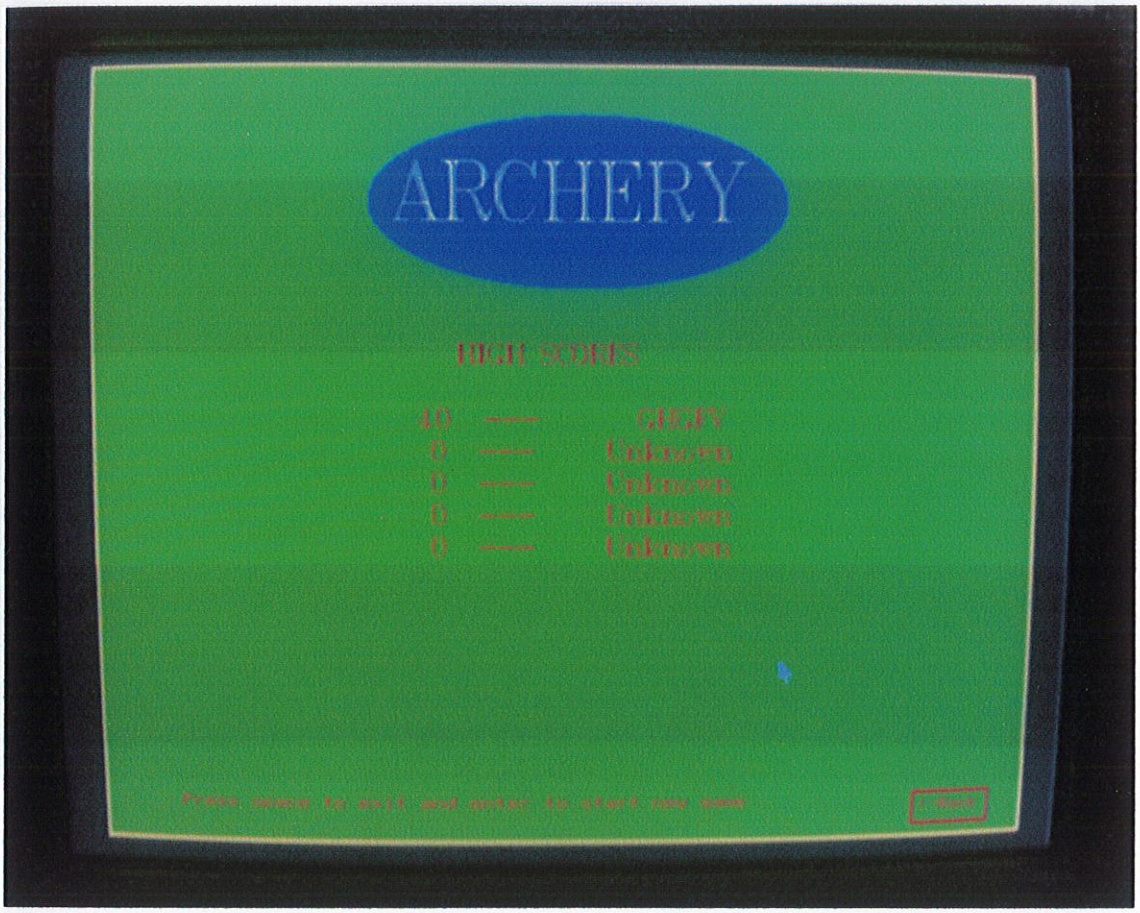
# Screen Shots

## The Menu Screen

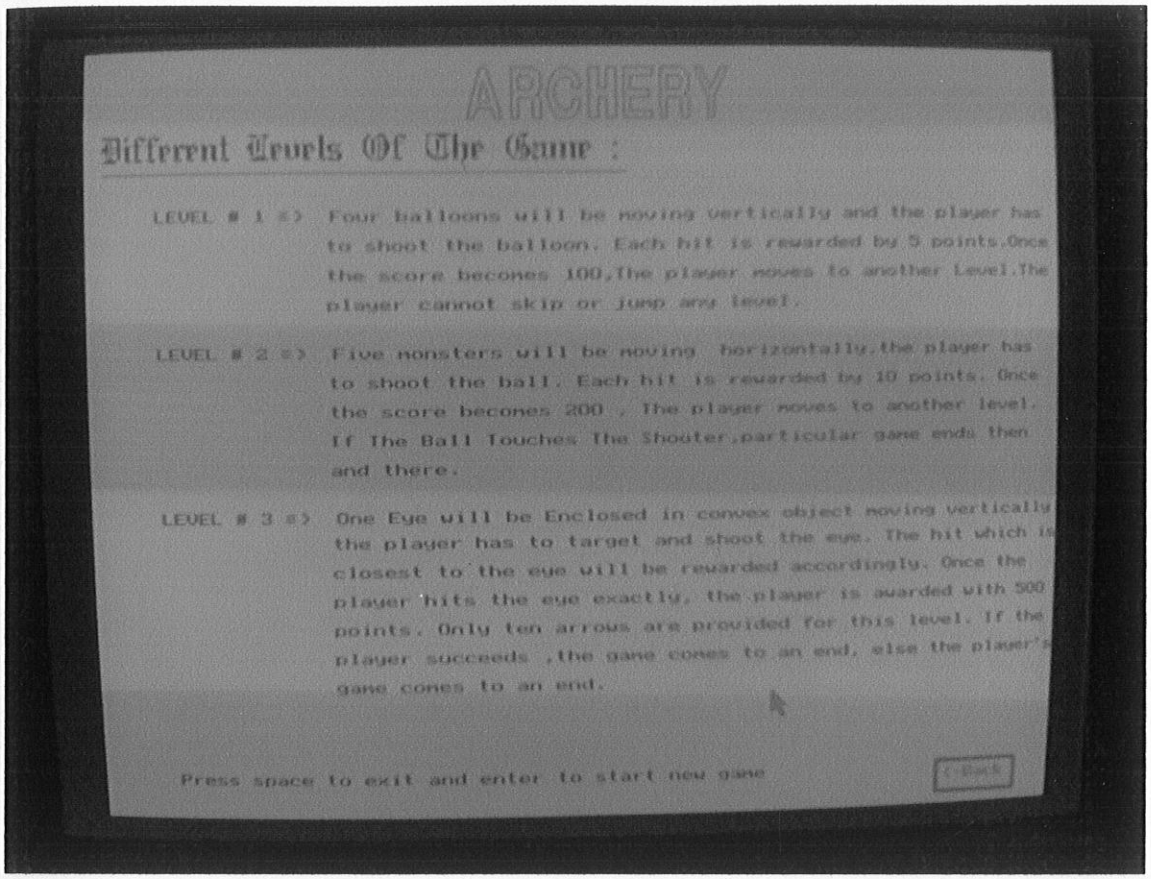
## Instructions



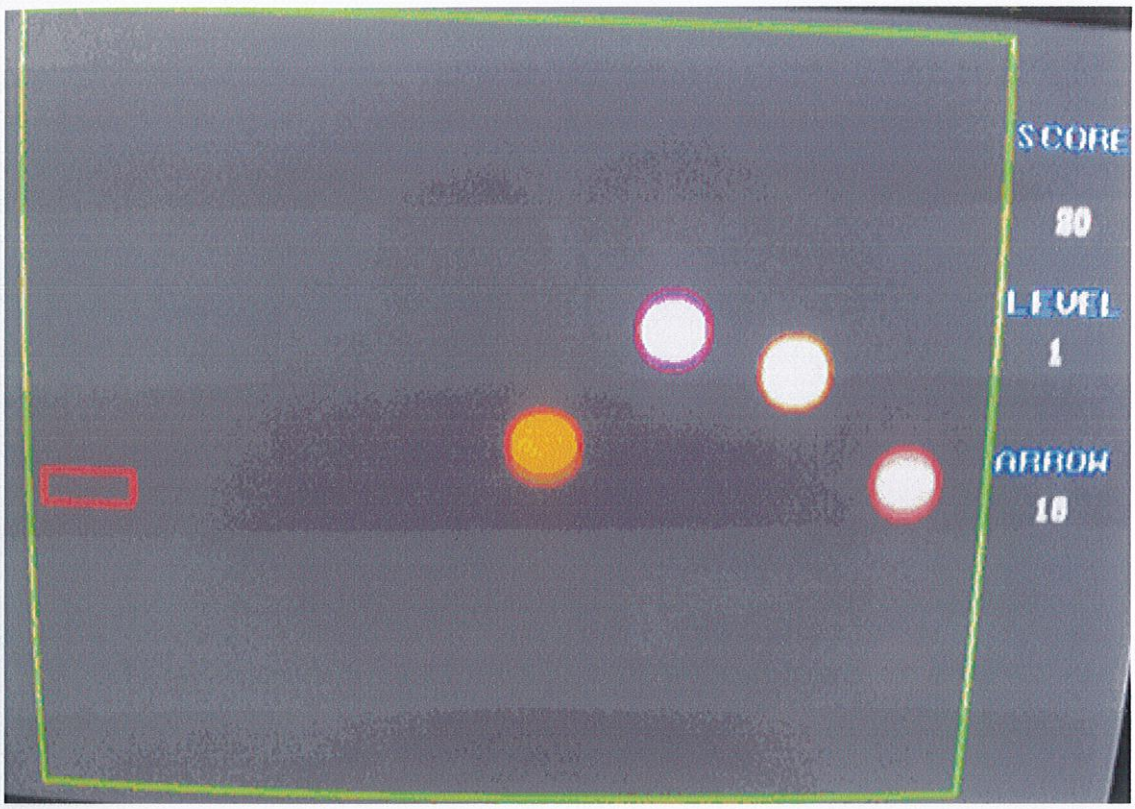
## High Scores

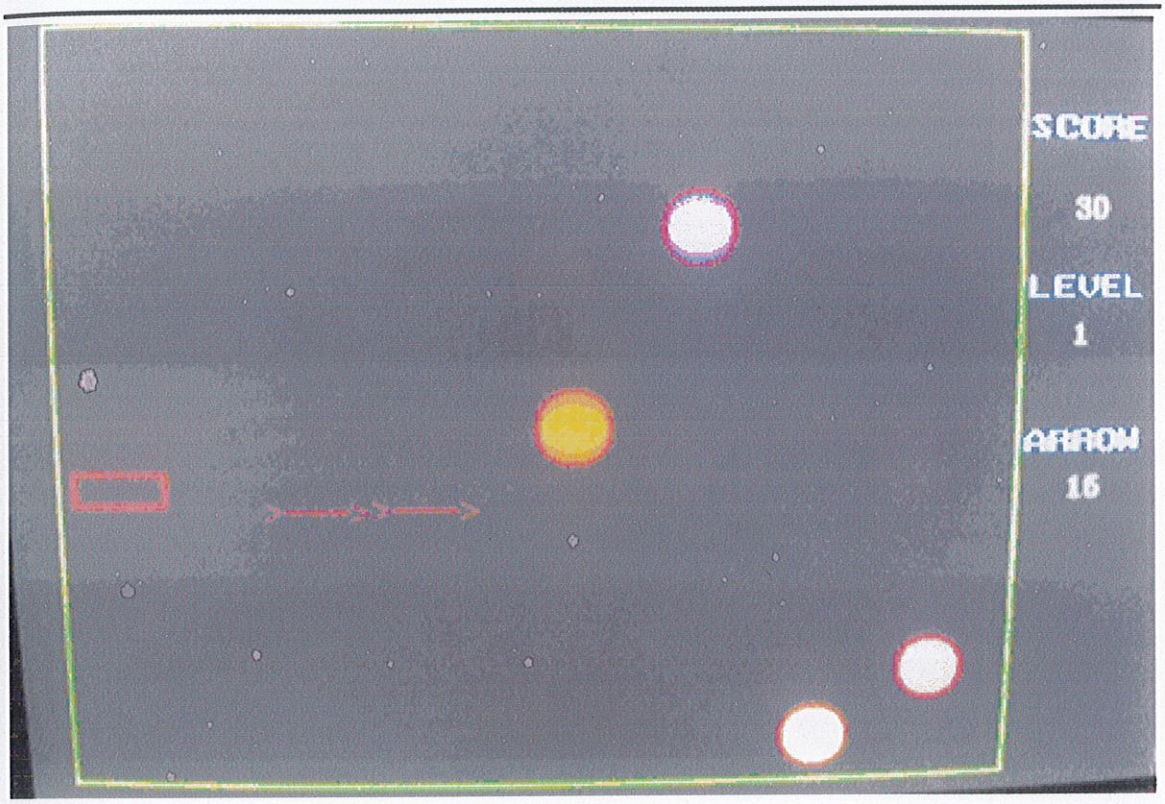

ARCHERY
HIGH SCORES
40 ---- GHGFV
0 ---- Unknown
0 ---- Unknown
0 ---- Unknown
0 ---- Unknown

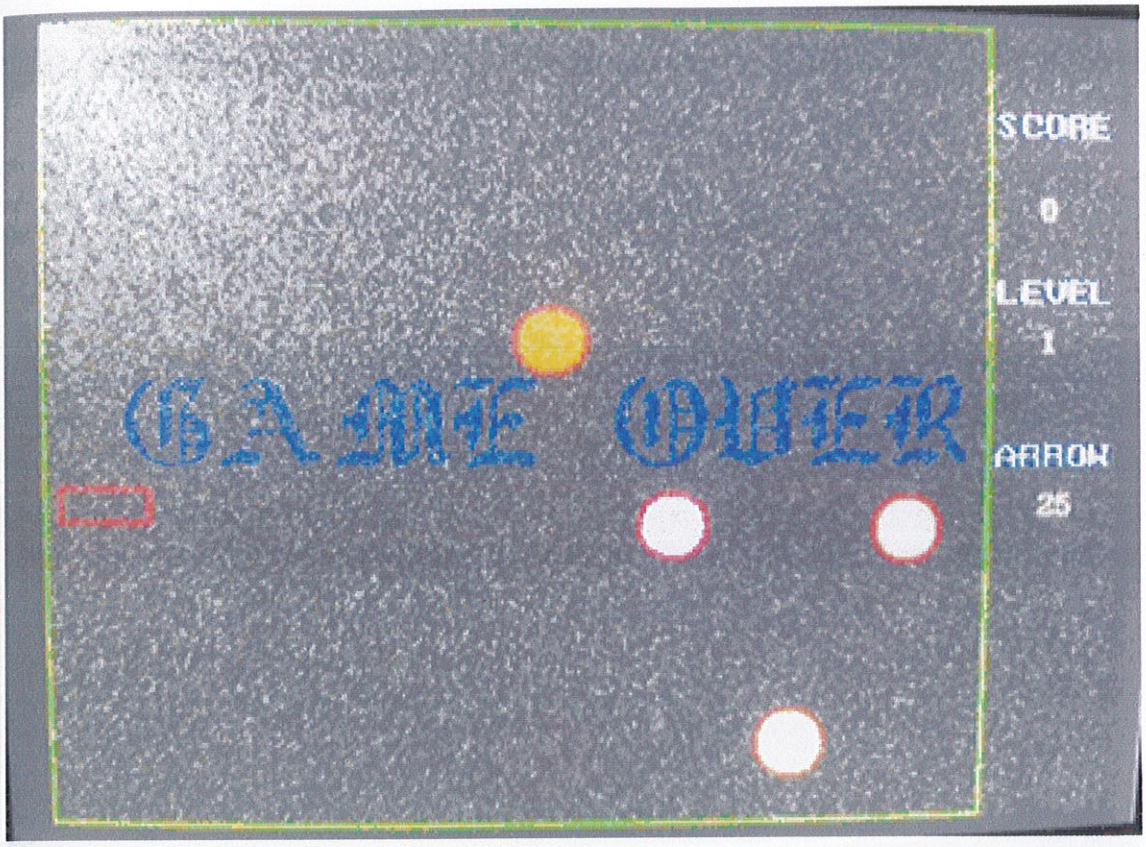
## Levels of Game



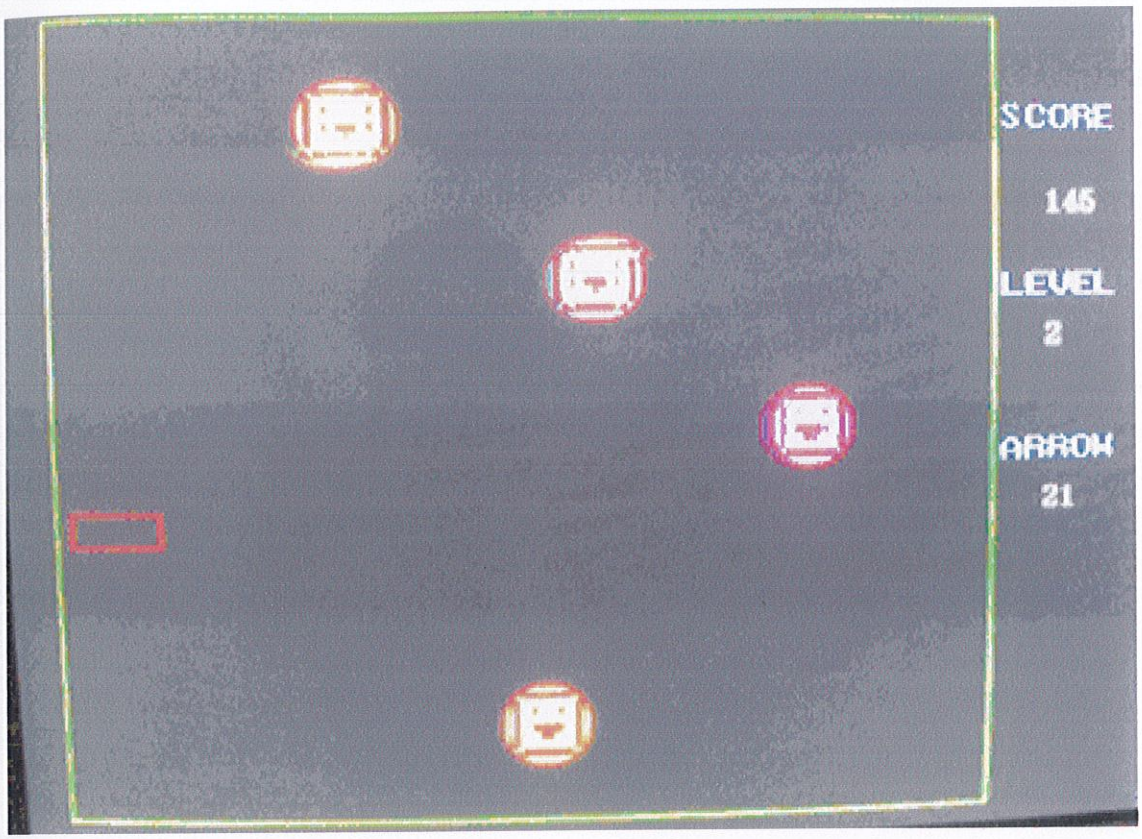
**Level 1**



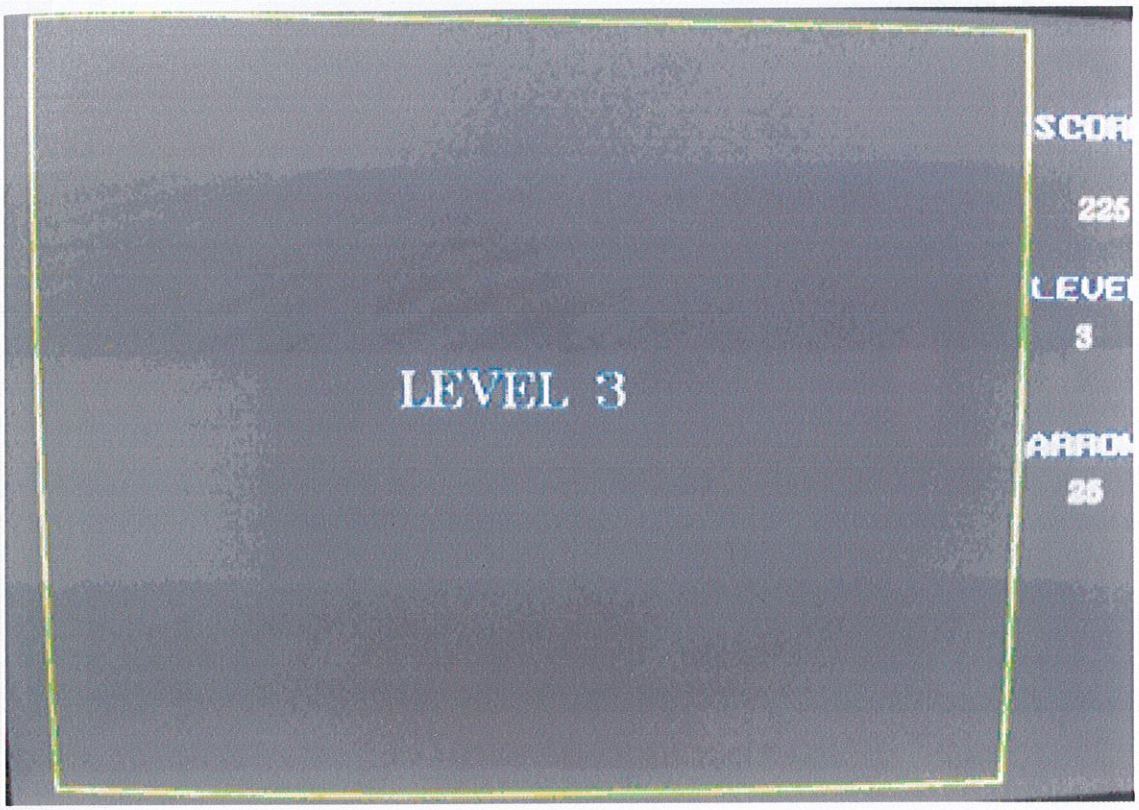

SCORE
30
LEVEL
1
ARROW
16


SCORE
0
LEVEL
1
GAME OVER
ARROW
25

**Level 2**



**Level 3**




WINNER
SCORE
755
LEVEL
3
ARROW
21

## Appendix A:
## Source code based on C

```
////////////////////////////////////////////////////
//////////////////  ARCHERY GAME  ///////////////
////////////////////////////////////////////////////

#include<conio.h>
#include<string.h>
#include<graphics.h>
#include<process.h>
#include<dos.h>
#include<stdio.h>
#include<stdlib.h>
//#include<math.h>

//structure declaration
struct scorechart
{
 char player[45];
 int scor;
};
struct scorechart s1[5];

//global variables
union REGS i,o;
char buff[80];
int hit=0,life=1;
int m1=1,m2=1,m3=1,m4=1,m5=1,gover=0;
int level1=1,check=0,score=0,mid,flag=0;
char buffer1[80],buffer[80];
```

```
int count=0,ascii,z1=625,scan,m=60,n=70,p=80,q=50,r;
int left=10,top=285,right=63,bottom=285;
unsigned int size,size1,size2,size3,size4,globalsize,handsize,eyesize;
void *pball,*pball1,*pball2,*pball3,*pball4,*handimage,*peye;
void *iball,*iball1,*iball2,*iball3;
int button,x,y;

//function declaration
void getkey();
void initmouse();
void showmouse();
void hidemouse();
void placemouse();
void getmousepos(int *,int *,int *);
void image1();
void image();
void hand();
void arrow(int);
int shoot(int);
void game();
void ball();
void eyelevel();
void balllevel2();
void gethighscores();
void displaytopscore();
void updatehighscore();
void writehighscorestodisk();
void rules();
void level();

void hand()//function to store hand image in memeory
```

```
{
setcolor(8);
setlinestyle(0,1,3);
setfillstyle(1,8);
bar(14,197,19,215);
line(19,198,23,198);
line(22,200,29,192);
line(29,192,38,192);
line(37,192,37,197);
line(32,197,54,197);
line(54,197,54,202);
line(32,202,54,202);
line(32,207,42,207);
line(32,212,42,212);
line(42,212,42,202);
line(38,212,38,217);
line(38,217,14,217);
handsize= imagesize(14,190,56,220);
/* allocate memory to hold the image */
handimage= malloc(handsize);
/* grab the image */
getimage(14,190, 56,220, handimage);
cleardevice();
}
void image1()//function to store level 1 balls in memeory
{
        unsigned int s,s1,s2,s3;
        setlinestyle(0,1,3);
        setcolor(RED);
        circle(500,60,20);
        setfillstyle(1,12);
```

```
        floodfill(500,60,RED);
        circle(430,70,20);
        setfillstyle(1,YELLOW);
        floodfill(430,70,RED);
        circle(360,80,20);
        setfillstyle(1,9);
        floodfill(360,80,RED);
        setcolor(RED);
        circle(290,50,20);
        setfillstyle(1,6);
        floodfill(290,50,RED);
        s = imagesize(470,35,530,85);        //k
        s1= imagesize(400,45,460,95);   //l
        s3= imagesize(330,55,390,105);//e
        s2= imagesize(260,25,320,75);     //f
        /* allocate memory to hold the image */
        iball= malloc(s);
        iball1=malloc(s1);
        iball2=malloc(s2);
        iball3=malloc(s3);
        /* grab the image */
        getimage(470,35, 530, 85, iball);
        getimage(400,45,460,95,iball1);
        getimage(330,55,390,105,iball2);
        getimage(260,25,320,75,iball3);
        putimage(470,35,iball,XOR_PUT);
        putimage(400,45,iball1,XOR_PUT);
        putimage(330,55,iball2,XOR_PUT);
        putimage(260,25,iball3,XOR_PUT);
}
void conveximage()//function to store eye image in memeory
```

```
{
        setlinestyle(0,1,3);
        setcolor(4);
        ellipse(500, 55, 0,360,13,35);
        setfillstyle(1,9);
        fillellipse(500,55,13,35);
        setcolor(0);
        putpixel(500,55,4);
        setlinestyle(0,1,1);
        ellipse(500, 55, 0,360,10.5,4.5);
        circle(500,55,4);
        eyesize=imagesize(480,15,520,95);
        peye=malloc(size);
        getimage(480,15,520,95,peye);
        putimage(480,15,peye,XOR_PUT);
}
void image()//function to store level 2 balls in memeory
{
        setlinestyle(0,1,3);
        setcolor(RED);
        circle(460,420,25);
        setfillstyle(1,YELLOW);
        floodfill(460,420,4);
        circle(470,330,25);
        setfillstyle(1,11);
        floodfill(470,330,RED);
        circle(480,240,25);
        setfillstyle(1,9);
        floodfill(480,240,RED);
        circle(490,150,25);
        setfillstyle(1,3);
```

```
floodfill(490,150,RED);
circle(500,60,25);
setfillstyle(1,10);
floodfill(500,60,RED);
settextstyle(0,0,5);
sprintf(buffer,"%c",1);
outtextxy(440,400,buffer);
outtextxy(450,310,buffer);
outtextxy(460,220,buffer);
outtextxy(470,130,buffer);
outtextxy(480,40,buffer);
size = imagesize(470,30,530,90);          //k
size1= imagesize(460,120,520,180);      //l
size2= imagesize(450,210,510,270);   //e
size3= imagesize(440,300,500,360);//f
size4= imagesize(430,390,490,450);
/* allocate memory to hold the image */
pball= malloc(size);
pball1=malloc(size1);
pball2=malloc(size2);
pball3=malloc(size3);
pball4=malloc(size4);
/* grab the image */
getimage(470,30, 530, 90,pball);
getimage(460,120,520,180,pball1);
getimage(450,210,510,270,pball2);
getimage(440,300,500,360,pball3);
getimage(430,390,490,450,pball4);
putimage(470,30,pball,XOR_PUT);
putimage(460,120,pball1,XOR_PUT);
putimage(450,210,pball2,XOR_PUT);
```

```
            putimage(440,300,pball3,XOR_PUT);
            putimage(430,390,pball4,XOR_PUT);
        }

        void rules()//function for showing the instructions
        {
          hidemouse();
          char buffer[80];
          cleardevice();
          setbkcolor(0);
          setcolor(4);
          setlinestyle(0,1,3);
          rectangle(0,0,640,480);
          setfillstyle(1,11);
          bar(3,3,637,477);
          setcolor(1);
          settextstyle(10,0,3);
          outtextxy(245,0,"ARCHERY");
          setcolor(8);
          settextstyle(4,0,3);
          outtextxy(15,60,"Rules To Be Followed : ");
          settextstyle(0,0,0);
          outtextxy(15,85,"_____");
          outtextxy(60,120, " Rule # 1 ð>  Only One Player Can Play The Game At A
Time.");
          outtextxy(60,150," Rule # 2 ð>  There Are Three Levels Of The game.");
          outtextxy(60,180," Rule # 3 ð>  Arrow Shooter Is Positioned By The ");
          sprintf(buffer," UP(%c) AND DOWN(%c) KEY.",24,25);
          outtextxy(444,180,buffer);
          outtextxy(60,210," Rule # 4 ð>  Only One Arrow Can Be Shooted At A
Time.");
```

```
        outtextxy(60,240," Rule # 5 ð>  SpaceBar Is Used To Shoot The Arrow.");
        outtextxy(60,270," Rule # 6 ð>  Enter Key Is Used To Select any Option.");
        outtextxy(60,300," Rule # 7 ð>  Five(5) Points Is Rewarded To Each Hit One
Makes.");
        outtextxy(60,330," Rule # 8 ð>  One Can Go To Another Level ,Crossing The
Previous Level.");
        outtextxy(60,360," Rule # 9 ð> Esc Key From The Main Will Terminate The
Game.");
        outtextxy(60,390," Rule # 10 ð> Exit Option Will Terminate The Game.");
         outtextxy(55,450,"Press space to exit and enter to start new game");
        setcolor(2);
        setfillstyle(1,11);
        rectangle(550,450,610,470);
        outtextxy(555,455,"<-Back");
        showmouse();
        do
        {
         getmousepos(&button,&x,&y);
         if((button&1)==1)
         {
          if((x>550)&&(x<610)&&(y>450)&&(y<470))
             break;
         }
        }while(!kbhit());
        hidemouse();
        cleardevice();
      }
      void level()
      {
        hidemouse();
        char buffer[80];
```

```
        cleardevice();
        setbkcolor(0);
        setcolor(4);
        setlinestyle(0,1,3);
        rectangle(0,0,640,480);
        setfillstyle(1,14);
        bar(3,3,637,477);
        setcolor(4);
        settextstyle(10,0,3);
        outtextxy(245,0,"ARCHERY");
        setcolor(8);
        settextstyle(4,0,3);
        outtextxy(15,45,"Different Levels Of The Game : ");
        settextstyle(0,0,1);
        outtextxy(15,70,"_____");
        outtextxy(50,100,"LEVEL # 1 ð>  Four balloons will be moving vertically and
the player has");
        outtextxy(55,118,"          to shoot the balloon. Each hit is rewarded by 5
points.Once");
        outtextxy(55,136,"          the score becomes 100,The player moves to another
Level.The");
        outtextxy(55,154,"          player cannot skip or jump any level.");
        outtextxy(50,184,"LEVEL # 2 ð>  Five monsters will be moving
horizontally,the player has");
        outtextxy(55,202,"          to shoot the ball. Each hit is rewarded by 10 points.
Once");
        outtextxy(55,220,"          the score becomes 200 , The player moves to
another level.");
        outtextxy(55,238,"          If The Ball Touches The Shooter,particular game
ends then");
        outtextxy(55,256,"          and there.");
```

```
        outtextxy(50,286,"LEVEL # 3 ð>  One Eye will be Enclosed in convex object
moving vertically");
        outtextxy(55,302,"          the player has to target and shoot the eye. The hit
which is ");
        outtextxy(55,320,"          closest to the eye will be rewarded accordingly.
Once the   ");
        outtextxy(55,338,"          player hits the eye exactly, the player is awarded
with 500  ");
        outtextxy(55,356,"          points. Only ten arrows are provided for this level. If
the  ");
        outtextxy(55,374,"          player succeeds ,the game comes to an end, else the
player's ");
        outtextxy(55,392,"          game comes to an end.");
        outtextxy(55,450,"Press space to exit and enter to start new game");
        setcolor(2);
        setfillstyle(1,11);
        rectangle(550,450,610,470);
        outtextxy(555,455,"<-Back");
        showmouse();
        do
        {
         getmousepos(&button,&x,&y);
         if((button&1)==1)
         {
          if((x>550)&&(x<610)&&(y>450)&&(y<470))
              break;
         }
        }while(!kbhit());
        hidemouse();
        cleardevice();
```

```
}
void updatehighscore()
{
        hidemouse();
        FILE *p;
        int i=0,t;
        struct scorechart s;
        clearviewport();
        setcolor(2);
        setcolor(2);
        setlinestyle(0,1,3);
        rectangle(0,0,550,478);
        setlinestyle(0,1,3);
        ellipse(280,80,0,320,133,50);
        setcolor(9);
        setfillstyle(1,9);
        fillellipse(280,80,133,50);
        setcolor(14);
        settextstyle(1,0,6);
        outtextxy(160,40,"ARCHERY");
        settextstyle(1,0,1);
        outtextxy(230,170,"SCORE CARD");
        setcolor(14);
        settextstyle(1,0,1);
        outtextxy(120,235,"Enter Your Name : ");
        for(i=0;i<=13;i++)
        {
         setcolor(i);
         settextstyle(1,0,1);
         outtextxy(250,440,"CONGRATULATIONS!!!!!!!");
         delay(100);
```

```
        if(i>=13)i=0;
        if(kbhit())break;
        }
        gotoxy(42,16);
        scanf("%s",s.player);
        s.scor=score;
        for(i=0;i<5;i++)
        {
            if(s.scor>s1[i].scor){t=i;break;}
        }
        for(i=4;i>t;i--)
        {
            strcpy(s1[i].player,s1[i-1].player);
            s1[i].scor=s1[i-1].scor;
        }
        strcpy(s1[t].player,s.player);
        s1[t].scor=s.scor;
        setcolor(14);
        outtextxy(250,320,"Press Any Key");
        getch();
}
void writehighscorestodisk()
{
        int i;
        FILE *p;
        p=fopen("ARCHERY.TXT","w");
        for(i=0;i<5;i++)
        fprintf(p,"%s\t%d\n",s1[i].player,s1[i].scor);
        fclose(p);
        return;
 }
```

```
//main begins
void main()
{
        int mid;
        static int c=1;
        char name[30];
        int gdriver = DETECT,gmode, errorcode;
        count=0; score=0;
        initgraph(&gdriver, &gmode, "c:\\tc\\bgi");
        errorcode = graphresult();
        if (errorcode != grOk)
        {
            printf("Graphics error: %s\n", grapherrormsg(errorcode));
            printf("Press any key to halt:");
            getch();
            exit(1); /* terminate with an error code */
        }
        initmouse();
        hand();
        image();
        image1();
        conveximage();
        cleardevice();
        int flag=0;

            while(1)
            {

              shine1:
              flag=0;
```

```
cleardevice();
setcolor(WHITE);
c=1;
setbkcolor(2);
setlinestyle(2,1,3);
rectangle(4,4,635,475);
setlinestyle(0,1,3);
rectangle(0,0,640,480);
setlinestyle(0,1,3);
ellipse(320,80,0,360,133,50);
setcolor(9);
setfillstyle(1,9);
fillellipse(320, 80, 133, 50);
setcolor(14);
settextstyle(4,0,6);
outtextxy(190,40,"ARChERY");
line(322,58,322,73);
settextstyle(0,0,3);
putimage(130,175,handimage,OR_PUT);
setcolor(4);
outtextxy(180,180,"PLAY THE GAME....");
setcolor(14);
outtextxy(180,230,"INSTRUCTIONS....");
outtextxy(180,280,"HIGH SCORES....");
outtextxy(180,330,"LEVELS....");
outtextxy(180,380,"EXIT....");
gethighscores();
showmouse();
do
{
 shine2:
```

```
getmousepos(&button,&x,&y);
if((button&1)==1)
{

 flag=1;
 if(x>180 && x<590 && y>180 && y<200)
    game();
 else if(x>180 && x<560 && y>230 && y<250)
    rules();
 else if(x>180 && x<540 && y>280 && y<300)
    displaytopscore();
 else if(x>180 && x<420 && y>330 && y<350)
    level();
 else if(x>180 && x<370 && y>380 && y<400)
    exit(0);
 else
    {goto shine2;}
}
if(kbhit())
{
 getkey();
 if(ascii==27)
    exit(0);
 else if(ascii==13)
 {
     switch(c)
     {
      case 1:game();break;
      case 2:rules();break;
      case 3:displaytopscore();break;
      case 4:level();break;
```

```
        case 5:exit(0);
        }
      goto shine1;
}
else if(scan==72)
{
    switch(c)//to remove the hand image
    {
     case 1:putimage(130,175,handimage,XOR_PUT);break;
     case 2:putimage(130,225,handimage,XOR_PUT);break;
     case 3:putimage(130,275,handimage,XOR_PUT);break;
     case 4:putimage(130,325,handimage,XOR_PUT);break;
     case 5:putimage(130,375,handimage,XOR_PUT);
    }
    if(c==1)
     c=5;
    else
     c--;
}
else if(scan==80)
{
    switch(c)//to remove the hand image
    {
     case 1:putimage(130,175,handimage,XOR_PUT);break;
     case 2:putimage(130,225,handimage,XOR_PUT);break;
     case 3:putimage(130,275,handimage,XOR_PUT);break;
     case 4:putimage(130,325,handimage,XOR_PUT);break;
     case 5:putimage(130,375,handimage,XOR_PUT);
}
    if(c==5)
     c=1;
```

```
                    else
                     c++;

                }

                switch(c)//to add the hand image
                {
                    case 1:putimage(130,175,handimage,OR_PUT);break;
                    case 2:putimage(130,225,handimage,OR_PUT);break;
                    case 3:putimage(130,275,handimage,OR_PUT);break;
                    case 4:putimage(130,325,handimage,OR_PUT);break;
                    case 5:putimage(130,375,handimage,OR_PUT);
                }
               }
              }while(flag==0);
            }//while
 }// main
void displaytopscore()
{
        hidemouse();
        FILE *p;
        int i;
        char str[100];
        p=fopen("ARCHERY.TXT","r");
        if(p==NULL)
        {
                for(i=0;i<5;i++)
                {
                    strcpy(s1[i].player,"Unknown");
                    s1[i].scor=0;
                }
```

```
}
else
{
for(i=0;i<5;i++)
        if(fscanf(p,"%s\t%d",s1[i].player,&s1[i].scor)==EOF)
        fclose(p);
}
fclose(p);
cleardevice();
setlinestyle(0,1,3);
rectangle(0,0,638,478);
setlinestyle(0,1,3);
ellipse(320,80,0,360,133,50);
setcolor(9);
setfillstyle(1,9);
fillellipse(320,80,133,50);
setcolor(14);
settextstyle(1,0,6);
outtextxy(200,40,"ARCHERY");
settextstyle(1,0,1);
setcolor(RED);
outtextxy(240,160,"HIGH SCORES");
setcolor(GREEN);
settextstyle(1,0,1);
for(i=0;i<5;i++)
{
        sprintf(str,"%7d  ---  %10s",s1[i].scor,s1[i].player);
        setcolor(4);
        outtextxy(170,200+(i*20),str);
}
settextstyle(0,0,1);
```

```
        setcolor(4);
        setfillstyle(1,11);
        rectangle(550,450,610,470);
        outtextxy(555,455,"<-Back");
         outtextxy(55,450,"Press space to exit and enter to start new game");
        showmouse();
        do
        {
         getmousepos(&button,&x,&y);
         if((button&1)==1)
         {
          if((x>550)&&(x<610)&&(y>450)&&(y<470))
           break;
         }
        }while(!kbhit());
        hidemouse();
        clearviewport();
} //end displaytopscore

void gethighscores()
{
        FILE *p;
        int i;
        char str[100];
        p=fopen("ARCHERY.TXT","r");
        if(p==NULL)
        {
           for(i=0;i<5;i++)
           {
                   strcpy(s1[i].player,"Unknown");
                   s1[i].scor=0;
```

```
            }
        }
        else
        {
            for(i=0;i<5;i++)
            if(fscanf(p,"%s\t%d\n",s1[i].player,&s1[i].scor)==EOF)
            fclose(p);
        }
        fclose(p);
    } //end gethighscore


void game()
{       hidemouse();
            level1=1;
            count=25;
            score=0;
            life=1;
            m1=1,m2=1,m3=1,m4=1,m5=1;
            cleardevice();
            setcolor(2);
            setlinestyle(0,1,3);
            rectangle(0,0,550,478);
            setbkcolor(0);
            setcolor(9);
            settextstyle(0,0,2);
            outtextxy(555,250,"ARROWS");
            outtextxy(555,50,"SCORE ");
            outtextxy(555,150,"LEVEL");
            setcolor(WHITE);
            settextstyle(1,0,1);
```

```
itoa(score,buff,5);
outtextxy(585,95,buff);
sprintf(buff,"%d",count);
outtextxy(585,275,buff);
itoa(level1,buffer,5);
setcolor(WHITE);
outtextxy(585,175,buffer);
setlinestyle(0,1,3);
setcolor(4);
rectangle(left,top-10,right,bottom+10);
while((1)&&(count!=0))
{


    if(score==100)
    {
        setcolor(0);
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        sprintf(buff,"%d",level1);
        outtextxy(585,175,buff);
        sprintf(buff,"%d",count);
        outtextxy(585,275,buff);
        score+=25;
        setfillstyle(1,0);
        bar(0,0,550,480);
        level1++,count=25;
        setcolor(2);
        setlinestyle(0,1,3);
        rectangle(0,0,550,478);
        settextstyle(1,0,1);
```

```
    setcolor(WHITE);
    sprintf(buff,"%d",count);
    outtextxy(585,275,buff);
    sprintf(buff,"%d",score);
    outtextxy(585,95,buff);
    sprintf(buff,"%d",level1);
    outtextxy(585,175,buff);
    settextstyle(1,0,4);
    for(int color=1;color<=14;color++)
    {
            setcolor(color);
            outtextxy(200,200,"LEVEL 2");
            delay(200);
    }
    setfillstyle(1,0);
    bar(200,100,400,300);
}
if(level1==1)
        ball();
if(level1==2&&score<200)
        balllevel2();
if(score==200)
{
    setcolor(0);
    sprintf(buff,"%d",score);
    outtextxy(585,95,buff);
    sprintf(buff,"%d",level1);
    outtextxy(585,175,buff);
    sprintf(buff,"%d",count);
    outtextxy(585,275,buff);
    score+=25;
```

```
        setfillstyle(1,0);
        bar(0,0,550,480);
        level1++,count=25;
        setcolor(2);
        setlinestyle(0,1,3);
        rectangle(0,0,550,478);
        settextstyle(1,0,1);
        setcolor(WHITE);
        sprintf(buff,"%d",count);
        outtextxy(585,275,buff);
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        sprintf(buff,"%d",level1);
        outtextxy(585,175,buff);
        settextstyle(1,0,4);
        for(int color=1;color<=14;color++)
        {
                setcolor(color);
                outtextxy(200,200,"LEVEL 3");
                delay(200);
        }
        setfillstyle(1,0);
        bar(200,100,400,300);
    }
    if(level1==3&&score>=225&&life==1)
            eyelevel();
    if(life==0)
       count=0;
    if(gover==1)
       count=0;
    if(kbhit())
```

```
        {
            getkey();
            if(ascii==27)
            break;
            if(scan==59)
            {
                showmouse();
                rules();
                setcolor(9);
                settextstyle(0,0,2);
                outtextxy(555,250,"ARROWS");
                outtextxy(555,50,"SCORE ");
                outtextxy(555,150,"LEVEL");
                setlinestyle(0,1,3);
                setcolor(4);
                rectangle(left,top-10,right,bottom+10);
            }
            if(scan==57)
            {
                mid=(top+bottom)/2;
                hit=1;
            }
            else if((scan==72)&&(z1>=510))
                        arrow(1);
            else if((scan==80)&&(z1>=510))
                        arrow(0);
        }
        if(hit==1)
                shoot(mid);
    } //while
    if(score<100)
```

```
                gover=1;
            if(score<500)
                gover=1;
            if(gover==1)
            {
                        setcolor(1);
                        settextstyle(4,0,8);
                        outtextxy(50,180,"GAME OVER");
                        for(int c=1,j=10;j<=32000;j++)
                        {
                            putpixel(random(j),random(j),8);
                            if(j==32000){j=0;c++;}
                            if(c>=50)break;
                        }
                        gover=0;
             }
            if(score>s1[4].scor)
                updatehighscore();
            else
                clearviewport();
            writehighscorestodisk();
            setcolor(4);
            rectangle(left,top-10,right,bottom+10);

            cleardevice();
            showmouse();
} //end function

void arrow(int pos)
{
        int s;
```

```
    //erase
    setfillstyle(SOLID_FILL,0);
    bar(left-5,top-15,right+5,bottom+15);
    if(pos==1)//up
    {
        if(top==60)
                top=60;
        else
        {
                top-=15;
                bottom-=15;
        }
    }
    else
    {
         if(top==420)
                top=420;
         else
         {
                top+=15;
                bottom+=15;
         }
    }
    //draw arrow
    setlinestyle(0,1,3);
    setcolor(4);
    rectangle(left,top-10,right,bottom+10);
    setlinestyle(0,1,1);
}

int shoot(int mid)
```

```
{
        static int z=90,delay_shoot=1;
        delay_shoot++;
        static int soundon=0;
        if(delay_shoot>200&&z<510)
        {
                delay_shoot=0;
                soundon=1;
                setcolor(0);
                if(soundon)
                 {
                 for(int Si=5; Si<10; Si++) {
                  sound(300*Si);
                  delay(1);
                  nosound();
                }}
                settextstyle(1,0,1);
                outtextxy(z-30,mid,">--->");
                setcolor(RED);
                outtextxy(z,mid,">--->");
                z+=30;
                z1=z;
        if(level1==1)
        {
                if((z>=480)&&(z<=500)&&(mid>=m-19)&&(mid<=m+19))
                {
                settextstyle(1,0,1);
                setcolor(0);
                sprintf(buff,"%d",score);
                outtextxy(585,95,buff);
                score+=5;
```

```
setcolor(WHITE);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
check=1;
}
else if((z>=410)&&(z<=430)&&(mid>=n-19)&&(mid<=n+19))
{
 setcolor(0);
 sprintf(buff,"%d",score);
 outtextxy(585,95,buff);
 score+=5;
 setcolor(WHITE);
 sprintf(buff,"%d",score);
 outtextxy(585,95,buff);
 check=2;
}
else if((z>=340)&&(z<=360)&&(mid>=p-19)&&(mid<=p+19))
{
setcolor(0);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
score+=5;
setcolor(WHITE);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
check=3;
}
else if((z>=270)&&(z<=290)&&(mid>=q-19)&&(mid<=q+19))
{

setcolor(0);
```

```
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            score+=5;
            setcolor(WHITE);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            check=4;
            }
    }
    if(level1==2)
    {
            if((abs(m-z)>=0)&&(abs(m-z)<=18)&&(mid==60))
            {
            setcolor(0);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            score+=5;
            setcolor(WHITE);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            check=1;
            setfillstyle(1,0);
            bar(z-50,30,z+70,100);
            z=500;
            m1++;
            }
           if((abs(n-z)>=0)&&(abs(n-z)<=17)&&(mid>=120)&&(mid<=150))
            {
            setcolor(0);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
```

```
score+=5;
setcolor(WHITE);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
check=2;
setfillstyle(1,0);
bar(z-50,100,z+90,180);
z=500;
m2++;
}
if((abs(p-z)>=0)&&(abs(p-z)<=18)&&(mid>=210)&&(mid<=240))
{
setcolor(0);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
score+=5;
setcolor(WHITE);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
check=3;
setfillstyle(1,0);
bar(z-50,180,z+80,285);
z=500;
m3++;
}
if((abs(q-z)>=0)&&(abs(q-z)<=19)&&(mid>=310)&&(mid<=340))
{
setcolor(0);
sprintf(buff,"%d",score);
outtextxy(585,95,buff);
score+=5;
```

```
            setcolor(WHITE);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            check=4;
            setfillstyle(1,0);
            bar(z-60,285,z+80,365);
            z=500;
            m4++;
            }
            if((abs(r-z)>=0)&&(abs(r-z)<=19)&&(mid>=410)&&(mid<=440))
            {
            setcolor(0);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            score+=5;
            setcolor(WHITE);
            sprintf(buff,"%d",score);
            outtextxy(585,95,buff);
            check=5;
            setfillstyle(1,0);
            bar(z-50,390,z+80,460);
            z=500;
            m5++;
            }
}
if(level1==3)
{
    if((z>=475)&&(z<=501)&&(mid>=m)&&(mid<=m+35))
    {
        settextstyle(1,0,1);
        setcolor(0);
```

```
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        score+=10;
        setcolor(WHITE);
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        check=1;
    }
    if((z>=475)&&(z<=501)&&(mid==m+30)&&(mid<=m+36))
    {
        settextstyle(1,0,1);
        setcolor(0);
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        score+=500;
        setcolor(WHITE);
        sprintf(buff,"%d",score);
        outtextxy(585,95,buff);
        setcolor(1);
        settextstyle(4,0,8);
        outtextxy(90,180,"WINNER");
        for(int c1=1,j=10;j<=25000;j++)
        {
                putpixel(random(j),random(j),8);
                if(j==32000){j=0;c1++;}
                if(c1>=50)break;
        }
        delay(1000);
        life=0;
        gover=0;
    }
```

```
            if((z>=475)&&(z<=501)&&(mid>=m+45)&&(mid<=m+65))
            {
                settextstyle(1,0,1);
                setcolor(0);
                sprintf(buff,"%d",score);
                outtextxy(585,95,buff);
                score+=10;
                setcolor(WHITE);
                sprintf(buff,"%d",score);
                outtextxy(585,95,buff);
                check=1;
            }
        }
        if(z>=510)
        {
                setcolor(0);
                outtextxy(z-30,mid,">--->");
                z=90;
                hit=0;
                setcolor(0);
                sprintf(buff,"%d",count);
                outtextxy(585,275,buff);
                count--;
                setcolor(WHITE);
                sprintf(buff,"%d",count);
                outtextxy(585,275,buff);
        }
    }    //end outer if
    return 0;
    }//end
```

```
void ball()
{
        static int k=35,delay_ball=1,l=45,e=55,f=25;
        delay_ball++;
        if(delay_ball>300)
        {
             delay_ball=0;
            //erase
            delay(20);
            setfillstyle(1,0);
            bar(470,k,530,k+50);
            bar(400,l,530,l+50);
            bar(330,e,530,e+50);
            bar(260,f,530,f+50);
            setcolor(2);
            setlinestyle(0,1,3);
            rectangle(0,0,550,478);
            if(check==1)
                  {
                    k=35;
                    check=0;
                  }
            if(check==2)
                  {
                    l=45;
                    check=0;
                  }
             if(check==3)
                  {
                    e=55;
                    check=0;
```

```
                }
                if(check==4)
                        {
                          f=25;
                          check=0;
                        }
                else if(k>=420)     k=35;
                else if(l>=420)     l=45;
                else if(e>=420)     e=55;
                else if(f>=420)    f=25;
                m=k+25; k+=8;
                n=l+25; l+=6;
                p=e+25; e+=7;
                q=f+25; f+=9;
                putimage(470,k,iball,OR_PUT);
                putimage(400,l,iball1,OR_PUT);
                putimage(330,e,iball2,OR_PUT);
                putimage(260,f,iball3,OR_PUT);
        }
} //end

void balllevel2()
{
    static int k=470,l=460,e=450,f=440,g=430,delay_ball=1;
    delay_ball++;
       //draw arrow
    if((top>=60)&&(top<=90)&&(m==62)&&(m1<=3)) {gover=1;return;}
    if((top>=120)&&(top<=180)&&(n==61)&&(m2<=3)){gover=1;return;}
    if((top>=210)&&(top<=270)&&(p==60)&&(m3<=3)){gover=1;return;}
    if((top>=295)&&(top<=365)&&(q==60)&&(m4<=3)){gover=1;return;}
    if((top>=380)&&(top<=465)&&(r==62)&&(m5<=3)){gover=1;return;}
```

```
if(delay_ball>300&&k>23&&l>17&&e>13&&f>10)
{
        delay_ball=0;
        delay(20);
        setfillstyle(1,0);
        if(m1<=3)     bar(k+60,30,k,90);
        if(m2<=3)     bar(l+60,120,l,180);
        if(m3<=3)     bar(e+60,210,e,270);
        if(m4<=3)     bar(f+60,300,f,360);
        if(m5<=3)     bar(g+60,390,g,450);
        setcolor(2);
        setlinestyle(0,1,3);
        rectangle(0,0,550,478);
        if(m1>=3)m=1000;
        if(m2>=3)n=1000;
        if(m3>=3)p=1000;
        if(m4>=3)q=1000;
        if(m5>=3)r=1000;
        if(m1<=3)
        {m=k,k-=8;}//k=8
        if(m2<=3)
        {n=l,l-=7;}   //l=7
        if(m3<=3)
        {p=e,e-=6;}     //e=6
        if(m4<=3)
        {q=f,f-=5;}       //f=5
        if(m5<=3)
        {r=g;g-=4;}          //g=4
        if(check==1)
        {
          k=470;
```

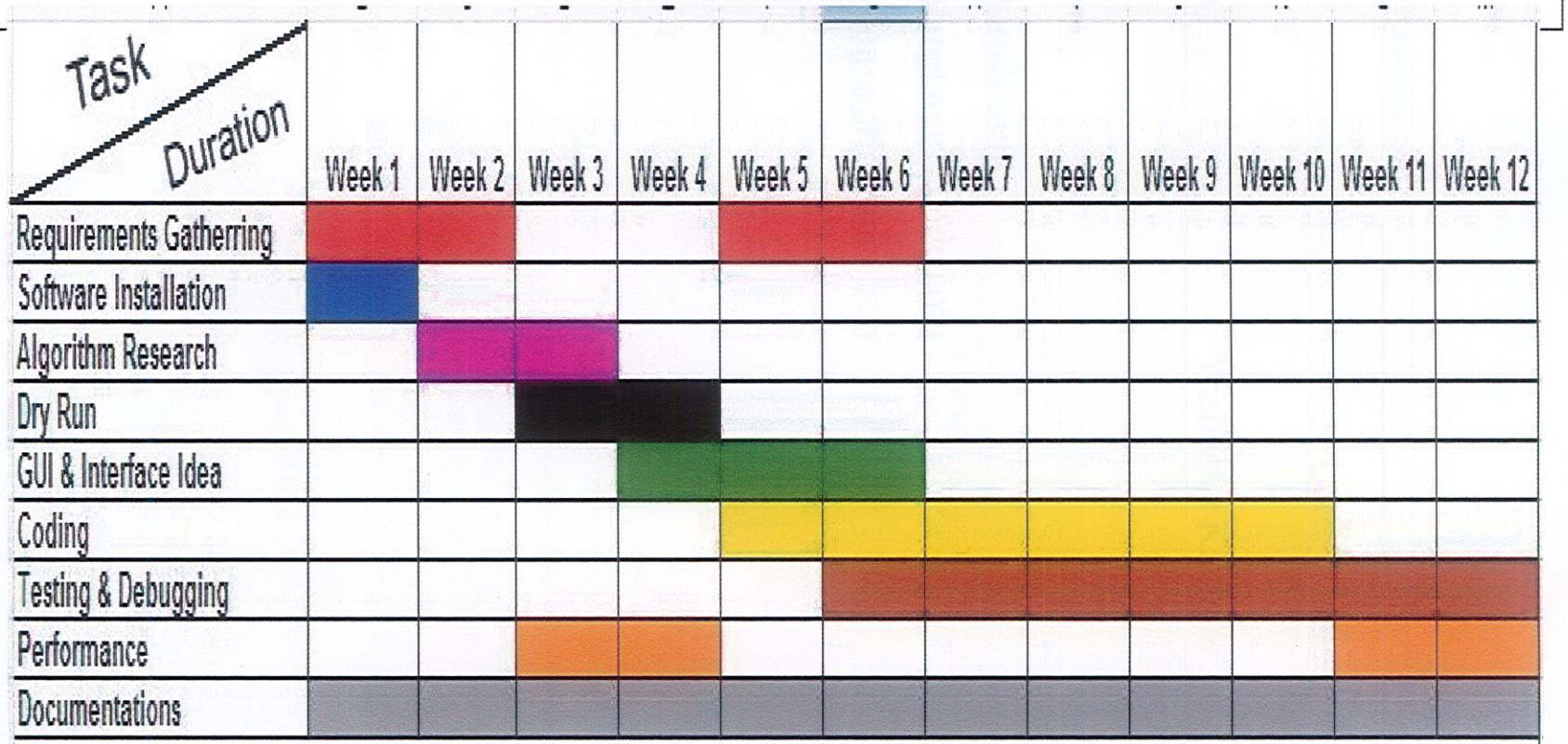
```
        check=0;
    }
    if(check==2)
    {
        l=460;
        check=0;
    }
    if(check==3)
    {
        e=450;
        check=0;
    }
    if(check==4)
    {
        f=440;
        check=0;
    }
    if(check==5)
    {
        g=430;
        check=0;
    }
    if (k<=23)     k = 470;
    if(l<=17)      l=460;
    if(e<=13)      e=450;
    if(f<=10)      f=440;
    if(g<=10)      g=430;
    //plot new image
    if(m1<=3)      putimage(k,30,pball,OR_PUT);
    if(m2<=3)      putimage(l,120,pball1,OR_PUT);
    if(m3<=3)      putimage(e,210,pball2,OR_PUT);
```

```
            if(m4<=3)      putimage(f,300,pball3,OR_PUT);
            if(m5<=3)      putimage(g,390,pball4,OR_PUT);
        }
    } //end

    void eyelevel()
    {
        static int e1=15,delay_eye=1;
        delay_eye++;
        static int up=1;
        if(delay_eye>300&&e1<405&&life!=0)
        {
            delay_eye=0;
            delay(20);
            setfillstyle(1,0);
            bar(480,e1,520,e1+80);
            setcolor(2);
            setlinestyle(0,1,3);
            rectangle(0,0,550,478);
            m=e1;
            if(up)
            e1+=5;
            if(up==0)
            e1-=5;
            if(check==1)
                check=0;
            if(e1>=405)
             {
                e1=400;
                up=0;
             }
```

```
            if(e1<=15)
              {
                  e1=15;
                   up=1;}
          putimage(480,e1,peye,OR_PUT);
  }
} //end
void getkey()
{
 i.h.ah=0x00;
 int86(0x16,&i,&o);
 ascii=o.h.al;
 scan=o.h.ah;
}

void initmouse()
{
 i.x.ax=0;
 int86(0x33,&i,&o);
}
void showmouse()
{
 i.x.ax=1;
 int86(0x33,&i,&o);
}
void hidemouse()
{
 i.x.ax=2;
 int86(0x33,&i,&o);
}
void getmousepos(int *button,int *x,int *y)
```

```
{
i.x.ax=3;
int86(0x33,&i,&o);
*button=o.x.bx;
*x=o.x.cx;
*y=o.x.dx;
}
```

# Appendix C:
# Gantt Chart




**Status : Project Completed**

# References

**Books:**

Let us C Yashwant Kanitkar

**Websites:**

www.google.com

www.planetsourcecode.com

www.vyomworld.com