“Implementation of LASER concept

in physics through animation”

CG mini-project

Submitted by:

Nishigandha Deore (SA23)

Aditya Kale (SA48)

Nishtha Jethani (SA68)

SECOND YEAR COMPUTER ENGINEERING

Under the Guidance of

Prof. Jyothishree R. K



Department of computer engineering

Hope Foundation’s

International Institute of Information Technology

Hinjewadi, Pune – 411057

AY 2022-2023

Semester-1

TABLE OF CONTENTS

1. PROBLEM STATEMENT

2. INTRODUCTION

3. ALGORITHM / PROGRAM

4. REQUIREMENTS

4.1 SOFTWARE AND HARDWARE DETAILS

4.2 LIBRARIES / PACKAGES USED

5. RESULT

6. CONCLUSION

7. REFERENCES

Problem Statement :

To display the concept of laser formation, atom excitation, photon movement, and related concepts with the light field in physics using the computer graphics that is the basic C graphics library. By making use of Turbo C++ & related system.

Introduction :

This is the project, based on computer graphics where using the concept of animation/ C++ graphics files (libraries). We have created the laser topic in physics. Laser generation is showcased, also the atom & electron excitation is showcased.

* In this project we have created 3 slides of animation using turbo C++.
* The initial slide is the introductory one. Where using “outtextxy” we have printed out the project name and the name of participants.
* The first starting slide/screen of the project amplification of photons.
* Second screen date comes after some delay of first screen is about the stimulation or excitation of atoms to release photons.
* The 3rd and the final screen is the device which is used to create lasers from one direction where voltage is applied and photons are released from the excitation of atoms hence creating the laser.

Program :

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<dos.h>

// FUNCTIONS USED

void starting();

void projectwork1();

void projectwork2();

void projectwork3();

void closing();

void main()

{

int gd=DETECT,gm,x,y;

initgraph(&gd,&gm,"c:\\tc\\bgi"); //Initializing Graphics Mode

starting();

projectwork1();

projectwork2();

projectwork3();

closing();

getch();

closegraph();

restorecrtmode();

}

void starting()

{

setbkcolor(BLUE);

setcolor(WHITE);

settextstyle(3,0,2);

outtextxy(100,50, "Welcome to our CG miniProject..");

outtextxy(100,80, "BY = 1) Aditya Kale");

outtextxy(105,100, "2) Nishtha Jethani");

outtextxy(105,120, "3) Nishigandha Deore");

outtextxy(100,150, "Our project is about photon, atom & related");

outtextxy(100,170, "concepts in physics");

delay(5000);

cleardevice();

}

void projectwork1() // AMPLIFICATION

{

setbkcolor(BLUE); // To set background color

setcolor(WHITE);

circle(170,230,8); // First object

// Different color filling & other uses..

setfillstyle(1,YELLOW);

floodfill(170,230,WHITE);

setcolor(GREEN);

settextstyle(11,0,2);

outtextxy(150,250,"PHOTON");

delay(2000);

// Setting text style...

settextstyle(7,0,4);

setcolor(WHITE);

outtextxy(110,380,"Amplification of Photons");

delay(1000);

setcolor(WHITE);

circle(250,180,8); //SECOND\_UP

circle(250,280,8);

floodfill(250,180,WHITE);

floodfill(250,280,WHITE);

delay(500);

circle(330,120,8); //THIRD\_UP

circle(330,195,8);

circle(330,265,8);

circle(330,340,8); //THIRD\_DOWN

floodfill(330,120,WHITE);

floodfill(330,195,WHITE);

floodfill(330,265,WHITE);

floodfill(330,340,WHITE);

delay(500);

circle(410,80,8);

circle(410,125,8);

circle(410,165,8);

circle(410,210,8);

circle(410,245,8);

circle(410,285,8);

circle(410,325,8);

circle(410,370,8);

floodfill(410,80,WHITE);

floodfill(410,125,WHITE);

floodfill(410,165,WHITE);

floodfill(410,210,WHITE);

floodfill(410,245,WHITE);

floodfill(410,285,WHITE);

floodfill(410,325,WHITE);

floodfill(410,370,WHITE);

delay(3000); // Given a delay time, so that there will be sufficient

// time in between execution of 2 functions...

// Then after that the screen gets cleared...

cleardevice();

}

void projectwork2()

{

setbkcolor(BLUE);

setcolor(WHITE);

setfillstyle(1,RED); // ATOM

circle(310,230,30);

floodfill(310,230,WHITE);

setcolor(YELLOW);

settextstyle(11,0,2);

outtextxy(296,270,"ATOM");

setcolor(WHITE);

setfillstyle(1,GREEN); // PHOTON

circle(190,110,5);

floodfill(190,110,WHITE);

setcolor(GREEN);

outtextxy(170,120,"PHOTON");

delay(2000);

setcolor(BLUE);

outtextxy(170,120,"PHOTON");

setfillstyle(1,BLACK);

floodfill(190,110,WHITE);

setcolor(BLACK);

circle(190,110,5);

setcolor(WHITE);

circle(210,130,5);

setfillstyle(1,GREEN);

floodfill(210,130,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(210,130,WHITE);

setcolor(BLACK);

circle(210,130,5);

setcolor(WHITE);

circle(230,150,5);

setfillstyle(1,GREEN);

floodfill(230,150,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(230,150,WHITE);

setcolor(BLACK);

circle(230,150,5);

setcolor(WHITE);

circle(250,170,5);

setfillstyle(1,GREEN);

floodfill(250,170,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(250,170,WHITE);

setcolor(BLACK);

circle(250,170,5);

setcolor(WHITE);

circle(270,190,5);

setfillstyle(1,GREEN);

floodfill(270,190,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(270,190,WHITE);

setcolor(BLACK);

circle(270,190,5);

setcolor(WHITE);

circle(295,215,5);

setfillstyle(1,GREEN);

floodfill(295,215,WHITE);

delay(300);

setfillstyle(1,RED);

floodfill(295,215,WHITE);

setcolor(RED);

circle(295,215,5);

setcolor(WHITE);

setfillstyle(1,YELLOW); // ATOM

circle(310,230,30);

floodfill(310,230,WHITE);

setcolor(YELLOW);

settextstyle(7,0,3);

outtextxy(115,50,"Stimulation of ATOM to Release Photons");

setcolor(WHITE);

delay(300);

setfillstyle(1,RED); // ATOM

circle(310,230,30);

floodfill(310,230,WHITE);

setcolor(WHITE);

circle(346,225,5);

setfillstyle(1,GREEN);

floodfill(346,225,WHITE);

circle(346,235,5);

floodfill(346,235,WHITE);

delay(800);

setfillstyle(1,BLACK);

floodfill(346,225,WHITE);

floodfill(346,235,WHITE);

setcolor(BLACK);

circle(346,225,5);

circle(346,235,5);

setcolor(WHITE);

circle(365,205,5);

setfillstyle(1,GREEN);

floodfill(365,205,WHITE);

circle(365,255,5);

floodfill(365,255,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(365,205,WHITE);

floodfill(365,255,WHITE);

setcolor(BLACK);

circle(365,205,5);

circle(365,255,5);

setcolor(WHITE);

circle(385,185,5);

setfillstyle(1,GREEN);

floodfill(385,185,WHITE);

circle(385,275,5);

floodfill(385,275,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(385,185,WHITE);

floodfill(385,275,WHITE);

setcolor(BLACK);

circle(385,185,5);

circle(385,275,5);

setcolor(WHITE);

circle(405,165,5);

setfillstyle(1,GREEN);

floodfill(405,165,WHITE);

circle(405,295,5);

floodfill(405,295,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(405,165,WHITE);

floodfill(405,295,WHITE);

setcolor(BLACK);

circle(405,165,5);

circle(405,295,5);

setcolor(WHITE);

circle(425,145,5);

setfillstyle(1,GREEN);

floodfill(425,145,WHITE);

circle(425,315,5);

floodfill(425,315,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(425,145,WHITE);

floodfill(425,315,WHITE);

setcolor(BLACK);

circle(425,145,5);

circle(425,315,5);

setcolor(WHITE);

circle(445,125,5);

setfillstyle(1,GREEN);

floodfill(445,125,WHITE);

circle(445,335,5);

floodfill(445,335,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(445,125,WHITE);

floodfill(445,335,WHITE);

setcolor(BLACK);

circle(445,125,5);

circle(445,335,5);

setcolor(WHITE);

circle(465,105,5);

setfillstyle(1,GREEN);

floodfill(465,105,WHITE);

circle(465,355,5);

floodfill(465,355,WHITE);

delay(300);

setfillstyle(1,BLACK);

floodfill(465,105,WHITE);

floodfill(465,355,WHITE);

setcolor(BLACK);

circle(465,105,5);

circle(465,355,5);

setcolor(WHITE);

circle(485,95,5);

setfillstyle(1,GREEN);

floodfill(485,95,WHITE);

circle(485,375,5);

floodfill(485,375,WHITE);

circle(190,110,5);

floodfill(190,110,WHITE);

setcolor(YELLOW);

settextstyle(11,0,2);

outtextxy(170,120,"PHOTON");

outtextxy(465,110,"PHOTON");

outtextxy(465,390,"PHOTON");

delay(5000);

cleardevice();

}

void projectwork3()

{

setbkcolor(BLACK);

setcolor(WHITE);

rectangle(200,150,460,160);

rectangle(200,300,460,310);

setfillstyle(1,GREEN);

floodfill(250,155,WHITE);

floodfill(250,305,WHITE);

ellipse(200,230,90,89,15,70); //MIRROR

setfillstyle(1,BLUE);

floodfill(205,232,WHITE);

ellipse(460,230,90,89,15,70); // SEMI SILVERED MIRROR

setfillstyle(1,CYAN);

floodfill(465,232,WHITE);

rectangle(300,365,360,380); // POWER SOURCE

setfillstyle(1,RED);

floodfill(305,370,WHITE);

moveto(300,365);

lineto(310,355);

lineto(370,355);

lineto(360,365);

moveto(360,380);

lineto(370,370);

lineto(370,355);

floodfill(305,362,WHITE);

floodfill(366,370,WHITE);

setcolor(YELLOW);

settextstyle(11,0,2);

outtextxy(288,390,"POWER SOURCE");

outtextxy(300,120,"ELECTRODES");

outtextxy(130,230,"MIRROR");

outtextxy(480,175,"SEMI SILVERED");

outtextxy(510,190,"MIRROR");

setcolor(WHITE);

moveto(370,363);

lineto(411,363);

lineto(411,265);

moveto(370,366);

lineto(413,366);

lineto(413,265);

rectangle(395,260,428,265);

floodfill(396,261,WHITE);

rectangle(395,200,428,205);

floodfill(396,201,WHITE);

rectangle(240,260,273,265);

floodfill(241,261,WHITE);

rectangle(240,200,273,205);

floodfill(241,201,WHITE);

moveto(255,265);

lineto(255,376);

lineto(300,376);

moveto(257,265);

lineto(257,373);

lineto(300,373);

setcolor(RED);

moveto(280,205);

lineto(280,260);

lineto(273,260);

moveto(273,205);

lineto(280,205);

moveto(435,205);

lineto(435,260);

lineto(428,260);

moveto(428,205);

lineto(435,205);

setcolor(WHITE);

circle(100,50,8);

setfillstyle(1,RED);

floodfill(100,50,WHITE);

circle(100,85,3);

setfillstyle(1,YELLOW);

floodfill(100,85,WHITE);

setcolor(YELLOW);

outtextxy(128,47,"ATOM");

outtextxy(128,82,"PHOTON");

// ATOM AND PHOTON MOVEMENT

setcolor(WHITE);

setfillstyle(1,RED); //ATOM

circle(365,205,8);

floodfill(365,205,WHITE);

circle(325,285,8);

floodfill(325,285,WHITE);

delay(3000);

circle(325,260,8);

floodfill(325,260,WHITE);

setfillstyle(1,BLACK);

floodfill(325,285,WHITE);

setcolor(BLACK);

circle(325,285,8);

setcolor(WHITE);

setfillstyle(1,RED);

circle(345,190,8);

floodfill(345,190,WHITE);

setfillstyle(1,BLACK);

floodfill(365,205,WHITE);

setcolor(BLACK);

circle(365,205,8);

delay(500);

setfillstyle(1,RED);

setcolor(WHITE);

circle(325,170,8);

floodfill(325,170,WHITE);

setfillstyle(1,BLACK);

floodfill(345,190,WHITE);

setcolor(BLACK);

circle(345,190,8);

setcolor(WHITE);

setfillstyle(1,YELLOW);

circle(250,250,3);

floodfill(250,250,WHITE);

delay(100);

circle(275,225,3);

floodfill(275,225,WHITE);

delay(100);

circle(295,205,3);

floodfill(295,205,WHITE);

delay(100);

circle(315,185,3);

floodfill(315,185,WHITE);

delay(100);

circle(320,205,3);

floodfill(320,205,WHITE);

delay(100);

circle(345,205,3);

floodfill(345,205,WHITE);

circle(365,205,3);

floodfill(365,205,WHITE);

delay(100);

circle(385,210,3);

floodfill(385,210,WHITE);

circle(395,220,3);

floodfill(395,220,WHITE);

circle(305,225,3);

floodfill(305,225,WHITE);

circle(325,245,3);

floodfill(325,245,WHITE);

delay(100);

circle(345,265,3);

floodfill(345,265,WHITE);

delay(100);

circle(345,235,3);

floodfill(345,235,WHITE);

delay(100);

circle(365,285,3);

floodfill(365,285,WHITE);

circle(365,230,3);

floodfill(365,230,WHITE);

delay(100);

circle(395,245,3);

floodfill(395,245,WHITE);

delay(100);

circle(415,225,3);

floodfill(415,225,WHITE);

delay(100);

circle(425,245,3);

floodfill(425,245,WHITE);

delay(300);

setlinestyle(SOLID\_LINE,1,3);

setcolor(YELLOW);

moveto(460,230);

lineto(600,230); //LASER

setcolor(YELLOW);

outtextxy(510,240,"LASER");

moveto(540,230);

lineto(535,225);

moveto(540,230);

lineto(535,235);

delay(3000);

cleardevice();

}

void closing()

{

setbkcolor(RED);

setcolor(YELLOW);

settextstyle(7,HORIZ\_DIR,4);

outtextxy(160,180,"!! THANK YOU !!");

outtextxy(120,280," Press any key to EXIT");

}

Requirements :

Software & Hardware Details :

Turbo C++ , C++ compiler (Min-GW), Windows 7 or higher PC

Libraries / Packages used :

1. stdio.h

2) conio.h

3) graphics.h

4) dos.h

(Header files/Libraries from C++ Libraries)

1. stdio.h :

The stdio.h header defines three variable types, several macros, and various functions for performing input and output.

1. Conio.h:

conio.h is a header file used for functions related to console input/output. conio.h has many inbuilt library functions that are used to perform input and output from a c program. Most C programs use this header file.

The functions are declared in conio.h depends on the compiler. As there exists more than one compiler each compiler has few functions declared in conio.hthat differs from other compilers. conio.hheader file that comes with Borland compiler of turbo C video RAM to perform Output functions. Few of the functions declared in conio.h are taken from Pascal.

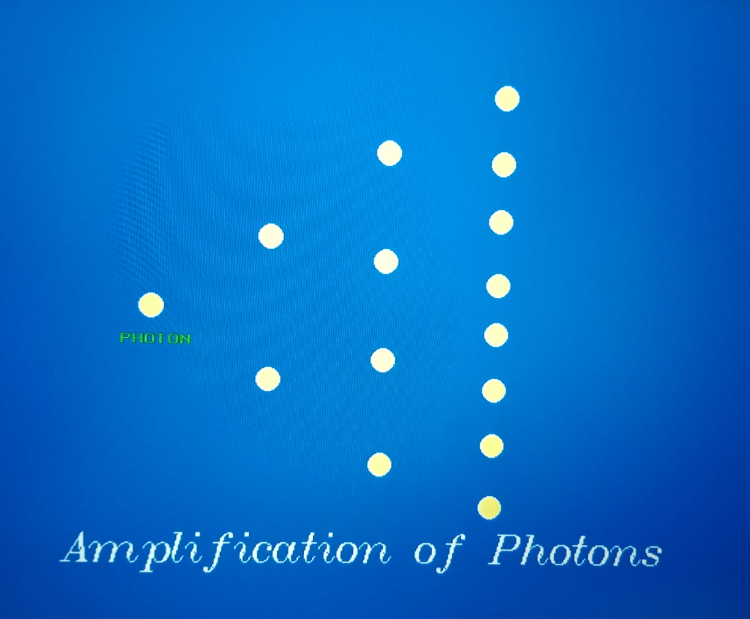
1. Graphics.h:

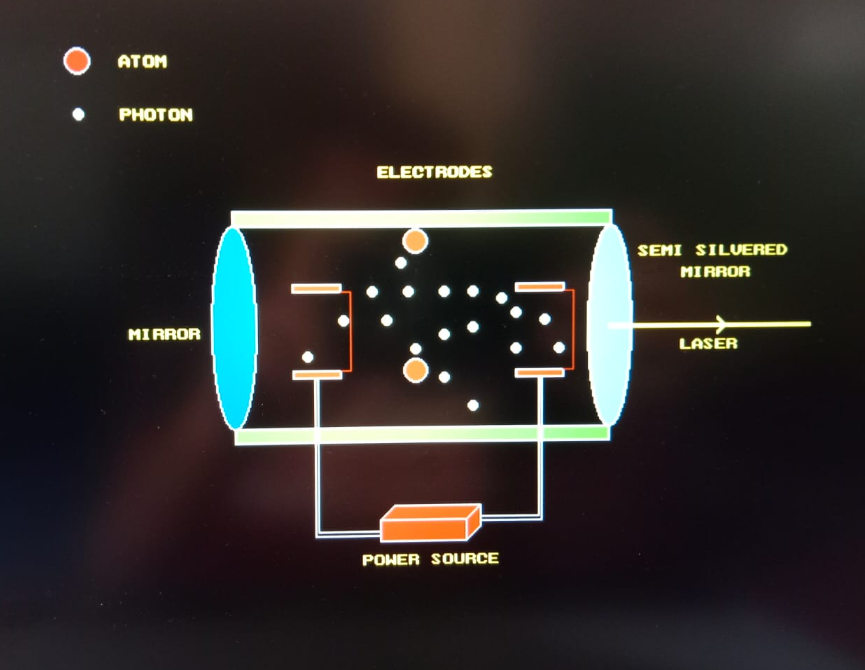
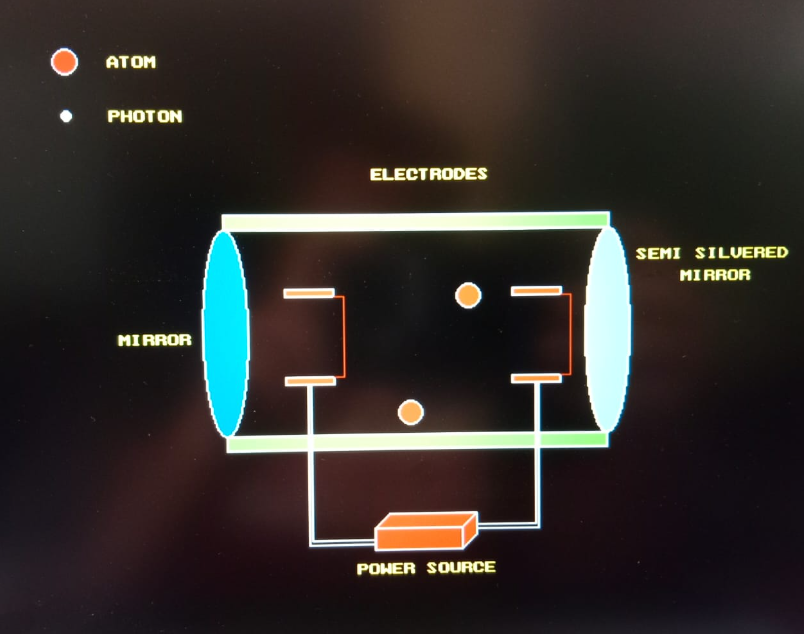
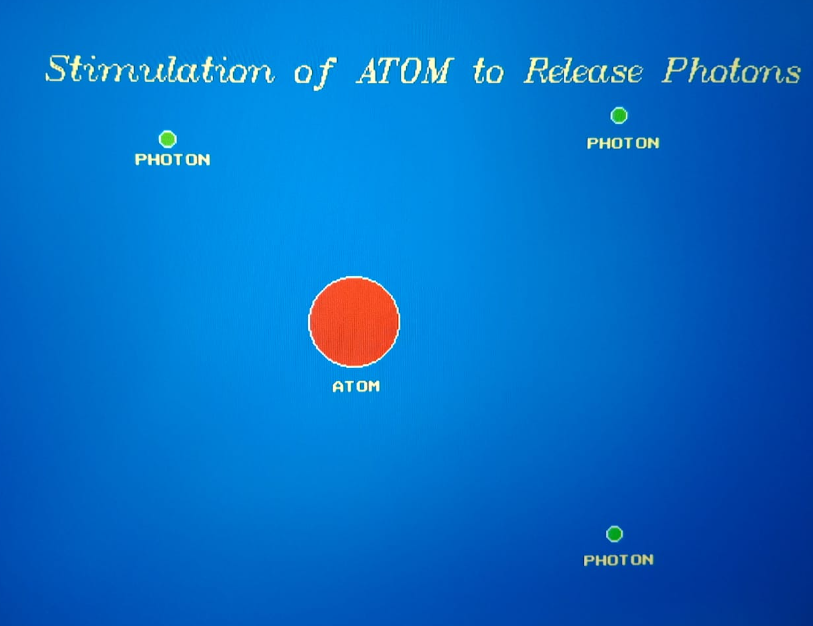
Graphics programming in C used to drawing various geometrical shapes (rectangle, circle, eclipse etc), use of mathematical function in drawing curves, coloring an object with different function in drawing an object with different colors & patterns & simple animation programs like jumping ball and moving cars.

1. dos.h :

dos.h header file of C language contains functions for handling interrupts, producing sound, date and time functions etc. It is Borland specific and works in Turbo C compiler.

Result :





Conclusion :

In this way we have successfully implemented the graphics libraries to make an interesting display / presentation of LASER concept in physics.

References :

C++ library documentation.

Laser concept notes in physics.

C++ graphics library function reference from internet.

THANK YOU !!!