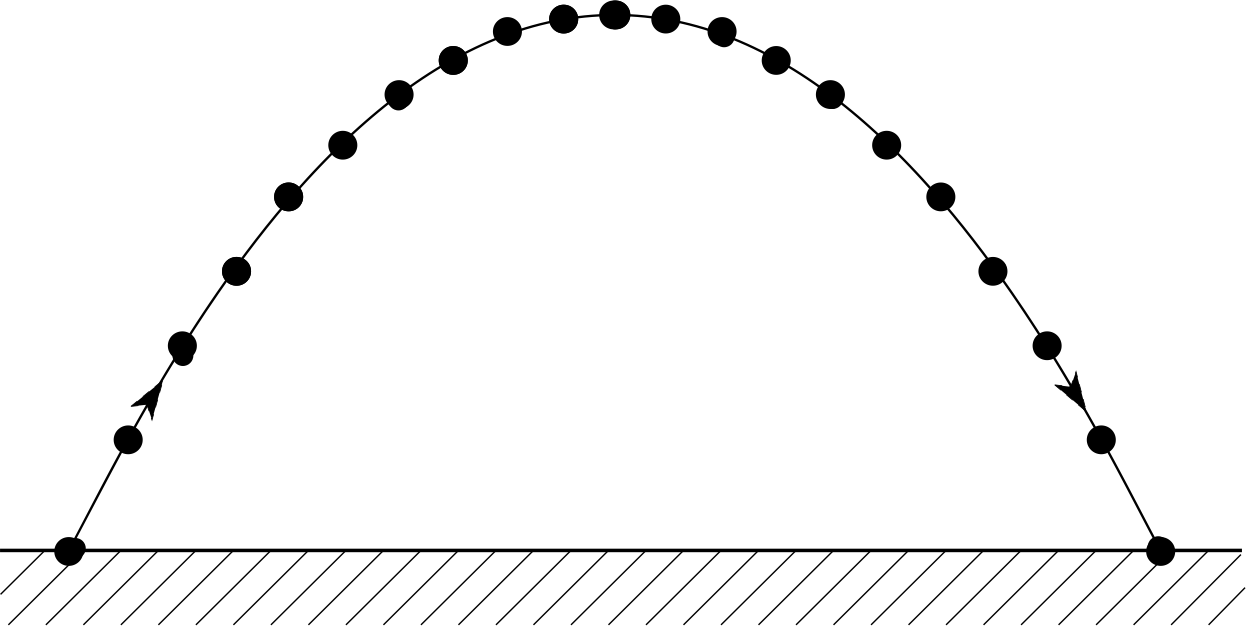
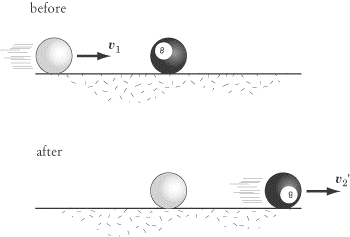
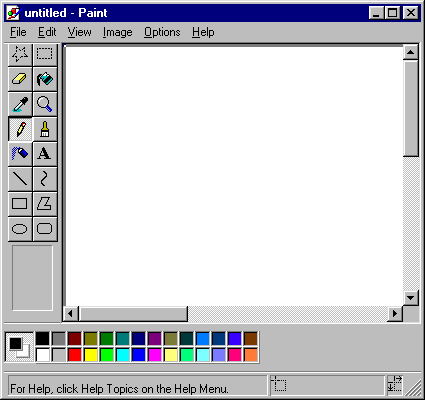
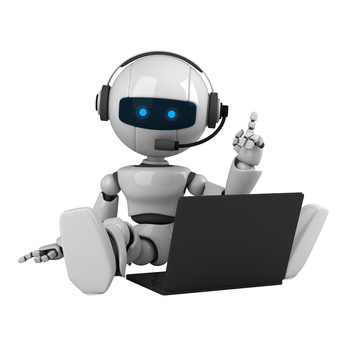
# Computer Project

**Projectile Motion Collisions**

  **Draw** **Chatbot**

-by Ujjawal Panchal

XII-A

Roll no. 5

Admn no. 3276

Certificate

This is to certify that Ujjawal Panchal of Class 12-A, has successfully completed his Computer Science Project.

Roll no.\_\_\_\_\_\_\_\_\_\_\_\_\_

Date:\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

Signature

Index

1. Certificate........................................................1
2. Acknowledgement..........................................2
3. Project code....................................................3
   1. Main code.................................................3
   2. Header file code.....................................19
4. Images...........................................................20

Project code

#include <animatio.h>

#include <ctype.h>

#include <fstream.h>

int fp=0, fc=0, fd=0, curr, in=0;

void Ltheme();

void tutp();

void switchf(int);

void collision();

void proj();

int ai();

int subqcmpi(char query[] , char str[]);

int querycmpi(char query[] , char str[]);

void drawfun();

class Nerve{

public :

char query[2000];

char reply[2000];

};

class draw

{

char ch;

int x, y;

public:

void drawf(char fname[]);

void tutd();

}objd;

void draw::drawf(char fname[])

{

int x=0, y=0;

char ch=0, op, name[10];

ofstream fx;

if(strcmpi(fname,"new.txt"))

{

ifstream fy(fname);

if(!fy)

{

cerr<<"Error.\nNow Creating New file";getch();

fy.close();delay(100);clrscr();return;

}

clrscr();

while(!fy.eof())

{

fy>>x;

fy>>y;

gotoxy(x, y);cout<<char(219);delay(10);

}

fy.close();

if(!strcmpi(fname, "welcome.txt"))return;

}

fx.open(fname, ios::ate);

do

{

op=getch();

switch(op)

{

case 'H':if(y!=1)

--y;

break;

case 'P':if(y!=24)

++y;

break;

case 'K':if(x!=1)

--x;

break;

case 'M':if(x!=80)

++x;

break;

case '1':ch=219;

break;

case '0':ch=0;

break;

}

gotoxy(x,y);

cout<<ch;

if(ch==char(219))

{

fx<<x<<" ";

fx<<y<<" ";

}

}

while(op!='e');

fx.close();

clrscr();

if(!strcmpi(fname, "new.txt"))

{

cout<<"Enter name of file:- ";gets(fname);

rename("new.txt", strcat(fname,".txt"));

remove("new.txt");

}

}

void draw::tutd()

{

int i;

clrscr();

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\n\t\t\t\tDraw Tutorial\n";

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\nThis programe allows you to draw. Use the arrow keys to move around.\n1 is pen down\n0 is pen up\ne is exit\n";

system("pause");

}

class coll

{

float u1, u2, m1, m2, v1, v2, e;

void calc();

public:

void getdata();

void collide();

void tutc();

}objc;

void coll::getdata()

{ cout<<"Enter mass of particle 1:- ";cin>>m1;

cout<<"Enter velocity of particle 1:- ";cin>>u1;

cout<<"Enter mass of particle 2:- ";cin>>m2;

cout<<"Enter velocity of particle 2:- ";cin>>u2;

e:

cout<<"Enter coefficient of restitution:- ";cin>>e;

if(e<0||e>1) goto e;

calc();

}

void coll::calc()

{

while(m1>10) m1/=10;

while(m2>10) m2/=10;

while(u1>10) u1/=10;

while(u2>10) u2/=10;

v1=((m1-(e\*m2))\*u1-((1-e)\*m2\*u2))/(m1+m2);

v2=((m2-(e\*m1))\*u2-((1-e)\*m1\*u1))/(m1+m2);

}

void coll::collide()

{

int t, tom, x1, x2, y=12;

char again;

tom=80/(u1+u2);

for(x1=0, x2=0, t=0; t<=tom; t++)

{

clrscr();

x1+=u1;x2+=u2;

gotoxy(x1,y);cout<<char(219);

gotoxy(80-x2,y);cout<<char(220);

delay(100);

}

while(x1>=0&&x2>=0 && x1<=80&&x2<=80)

{

clrscr();

x1+=v1;x2+=v2;

gotoxy(x1,y);cout<<char(219);

gotoxy(80-x2,y);cout<<char(220);

delay(200);

}

clrscr();

cout<<"Final velocities:-\nv1="<<v1<<"\nv2=:-"<<v2;

cout<<"\nAgain?Y/N:- ";cin>>again;

if(again=='Y' || again=='y')collide();

}

void coll::tutc()

{

int i;

clrscr();

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\n\t\t\t\tCollision Tutorial\n";

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\nThis programe allows you to observe collisions in 1 dimension\n";

system("pause");

}

class page

{

char arr[24][60], fname[10], noflines;

public:

int active;

page(char a[1]);

void print();

void pagel(int i, int j);

void pager(int i, int j);

void marque(){marquee(arr, noflines);}

void ani(){anim(arr,noflines);}

} p[3]={"a", "b", "c"};;

page::page(char a[1])

{

int i=0;

strcpy(fname,"f");

strcat(fname,a);

ifstream fin;

fin.open(fname, ios::binary);

while(!fin.eof())

{

fin.read((char\*)&arr[i], 80);

i++;

}

noflines=--i;

}

void page::print()

{

int i;

for(i=0; i<noflines; i++)

cout<<arr[i]<<endl;

}

void page::pager(int i, int j) //go right screen

{

int k;

if(active==1)

{

for(k=j; k<80; k++)

{

cout<<arr[i][k];

}

}

else

{

for(k=80-j; k<80; k++)

{

cout<<arr[i][k-(80-j)];

}

}

}

void page::pagel(int i, int j) //go left screen

{

int k;

if(active==1)

{

for(k=j; k<80; k++)

{

cout<<arr[i][k-j];

}

}

else

{

for(k=80-j; k<80; k++)

{

cout<<arr[i][k];

}

}

}

class projectile

{

float vx, vy, t, i, v, angle, tof, range, height;

int x, y;

void calc();

public:

void getvelp()

{

cout<<"Enter velocity of projection:- ";cin>>v;

cout<<"Enter angle of projection:- ";cin>>angle;

calc();

}

void dispp();

void tutp();

}objp;

void projectile::tutp()

{

int i;

clrscr();

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\n\t\t\t\tProjectile Tutorial\n";

for(i=0; i<79; i++)

cout<<char(219);

cout<<"\nThis programme animates the path taken by a projectile launched at an input velocity and angle.\n\nIt also scales the input value suitably to display path for a wide range of input velocities.\n\n";

system("pause");

}

void projectile::calc()

{

while(v>=10)

v/=10;

v\*=5;

vx=v\*cos((angle/180)\*M\_PI);

vy=v\*sin((angle/180)\*M\_PI);

tof=(2\*vy)/9.8;

range=vx\*tof;

height=(vy\*vy)/(2\*9.8);

}

void projectile::dispp()

{

clrscr();

char again;

for(i=0; i<80; i++)

{ gotoxy(i, 24);cout<<'\_'; }

for(x=0, y=0, i=0; i<=tof;i++, x+=vx, y+=vy-(9.8\*i))

{

gotoxy(x+1,24-y); cout<<char(-37);

gotoxy(1,1);cout<<x<<' '<<y<<' '<<' '<<tof;

delay(200);

}

gotoxy(range, 24);cout<<char(219);

getch();

clrscr();

cout<<"Time of Flight:- "<<tof<<"\nMaximum Height:- "<<height<<"\nRange:- "<<range<<endl;

cout<<"\nAgain?Y/N:- ";cin>>again;

if(again=='Y'||again=='y')

dispp();

}

int main()

{

clrscr();

if (in==0)

{

draw d;d.drawf("welcome.txt");

delay(1000);

clrscr();

}

char arra[][80]={" ", " ", " ", " ", " ", " "}, arrb[][80]={"\t\tPhysics", "1.Projectile", "2.Collisions","3.Exit", "Enter choice:- "},arrc[][80]={"\t\tMiscellaneous", "1.Draw", "2.L's Theme", "3.Exit", "Enter choice:- ", "\n "};

ofstream fa("fa", ios::binary);

ofstream fb("fb", ios::binary);

ofstream fc("fc", ios::binary);

for(int i=0; i<6; i++)

{

fa.write((char\*)&arra[i],80);

fb.write((char\*)&arrb[i],80);

fc.write((char\*)&arrc[i],80);

}

char ch;

int j, k, max;

curr=p[1].active=1; max=2;

do

{

clrscr();

p[curr].print();

ch=getch();

if(ch=='M' && curr>0)

{

for(j=0; j<80; j++)

{

clrscr();

for(i=0; i<7; i++)

{

p[curr-1].pagel(i, j);

p[curr].pagel(i, j);

}

delay(10);

}

p[curr].active=0;curr--;p[curr].active=1;

clrscr();

p[curr].print();

}

else if(ch=='K' && curr<max)

{

for(j=0; j<80; j++)

{

clrscr();

for(i=0; i<7; i++)

{

p[curr].pager(i, j);

p[curr+1].pager(i, j);

}

delay(10);

}

p[curr].active=0;curr++;p[curr].active=1;

clrscr();

p[curr].print();

}

else if(ch<='3' && ch>='1')

switchf((curr\*10)+(ch-'1')+1);

if(curr==0){ai();curr=1;}

}while (ch!='e' && ch!='E' && ch!='4');

return 0;

}

void switchf(int a)

{

clrscr();

switch(a)

{

case 11:proj();

break;

case 12:collision();break;

case 21:drawfun();break;

case 22:Ltheme();break;

case 13:

case 23:exit(0);

}

p[curr].print();

return;

}

void proj()

{

int i, choice;

if(fp==0)

{

objp.tutp();

fp=1;

}

do

{

char str[5][60]={"Welcome to Projectile, select an option:-", "1.Projectile", "2.Tutorial", "3.Go Back"};

anim(str, 4);

do

{

cout<<"\nEnter choice:- ";

cin>>choice;

if(choice>5 || choice<1)

cout<<"Invalid choice\n";

}

while(choice>5 || choice<1);

marquee(str, 4);

switch(choice)

{

case 1: clrscr();

objp.getvelp();

objp.dispp();

clrscr();

break;

case 2:objp.tutp();

break;

}

}

while(choice!=3);

}

void collision()

{

clrscr();

int choice;

if(fc==0)

{

objc.tutc();

fc=1;

}

do

{

char str[4][60]={"Welcome to collisions","1.Collision","2.Tutorial", "3.Go back"};

anim(str, 4);

do

{ cout<<"\nEnter choice:- ";cin>>choice;

if(choice<1||choice>3)cout<<"\nInvalid choice";

}while(choice<1||choice>3);

marquee(str, 4);

switch(choice)

{

case 1:objc.getdata();

objc.collide();

break;

case 2:objc.tutc();

break;

}

}while(choice!=3);

}

void drawfun()

{

clrscr();

int choice;

char fname[20];

if(fd==0)

{

objd.tutd();

fd=1;

}

do

{

char str[5][60]={"Welcome to Drawing", "1.New", "2.Open", "3.Tutorial", "4.Go Back"};

anim(str, 5);

do

{

cout<<"\nEnter choice:- ";

cin>>choice;

if(choice>5 || choice<1)

cout<<"Invalid choice\n";

}while(choice>5 || choice<1);

marquee(str, 5);

switch(choice)

{

case 1:objd.drawf("new.txt");remove("new.txt");

break;

case 2:cout<<"Enter name of file:- ";gets(fname);

objd.drawf(strcat(fname, ".txt"));

break;

case 3:objd.tutd();break;

}

}while(choice!=4);

}

int subqcmpi(char query[] , char str[])

{

int i,j,match=1;

char temp1[200]={'\0'};char temp2[200]={'\0'};

for(i=0,j=0;query[i]!='\0';i++)

{

if(isalnum(query[i])!=0)

{

temp1[j]=query[i];

j++;

}

}

if(isalnum(temp1[j])==0)temp1[j]='\0';

for(i=0,j=0;str[i]!='\0';i++)

{

if(isalnum(str[i])!=0)

{

temp2[j]=str[i];

j++;

}

}

if(isalnum(temp2[j])==0)temp2[j]='\0';

for(j=0;temp1[j]!=0;j++)

if(temp1[j]!=temp2[j])match=0;

if(match==1)return 0;

else return 1;

}

int querycmpi(char query[] , char str[])

{

int i,j;

char temp1[200]={'\0'};char temp2[200]={'\0'};

for(i=0,j=0;query[i]!='\0';i++)

{

if(isalnum(query[i])!=0)

{

temp1[j]=query[i];

j++;

}

}

if(isalnum(temp1[j])==0)temp1[j]='\0';

for(i=0,j=0;str[i]!='\0';i++)

{

if(isalnum(str[i])!=0)

{

temp2[j]=str[i];

j++;

}

}

if(isalnum(temp2[j])==0)temp2[j]='\0';

if(strcmpi(temp1,temp2)==0)return 0;

else return 1;

}

int ai()

{

Nerve o; char repu[2000],query[200];

int found=0,i;

Nerve p;

do

{

GETQ:

fstream brain("Brain",ios::binary|ios::app|ios::in);

//----------------------------getting query and dropdown list------------//

clrscr();

char qu[2000]={'\0'}; i=0;int j=0,fscan=0;Nerve n;

qu[0]='~';

do

{

if(kbhit())

{

brain.close();brain.open("Brain",ios::binary|ios::app|ios::in);

int match=0;j=0;

char rep[10][200]={'\0'};

if(fscan!=0)i++;

clrscr();

qu[i]=getch();

if(qu[i]=='K')return 0; if(qu[i]==13)break;

while(brain.read((char\*)&n,sizeof(n)))

{

if(subqcmpi(qu,n.query)==0)

{

for(int k=0;n.query[k]!='\0';k++)rep[j][k]=n.query[k];

j++;

match=1;

}

}

cout<<qu<<endl;

if(match==1)

{

for(;j>=0;j--)cout<<rep[j]<<endl;

}

fscan=1;

}

}while((qu[i]!='\0'));

clrscr();

//----------------------------getting query and dropdown list------------//

if(querycmpi(qu,"cmd/cls")==0 || querycmpi(qu,"cls")==0)

{

clrscr();

goto GETQ;

}

found=0;

if(querycmpi(qu,"cmd/forget")==0 || querycmpi(qu,"forget")==0)

{

char choice;

ofstream fout("Brain2",ios::binary);

cout<<"What do you want me to forget? :>";gets(qu);

cout<<"Are you sure ? (y/n) :>";cin>>choice;

if(choice=='y')

{

while(brain)

{

brain.read((char\*)&o,sizeof(o));

if(querycmpi(qu,o.query)==0);

else

fout.write((char\*)&o,sizeof(o));

}

brain.close();

fout.close();

remove("Brain");

rename("Brain2","Brain");

cout<<"Query has been forgotten successfully!"<<endl;

}

else if(choice=='n')cout<<"Query has not been forgotten! "<<endl;

else cout<<"Command Overruled!"<<endl;

goto GETQ;

}

if((querycmpi(qu,"L's theme")==0) || (querycmpi(qu,"Ltheme")==0))found=2;//directs found to L'stheme.

else if(querycmpi(qu,"collision")==0)found=3;//directs found to collision.

else if(querycmpi(qu,"projectile")==0)found=4;//directs found to projectile.

else if((querycmpi(qu,"draw")==0) || (querycmpi(qu,"drawing")==0))found=5;//directs found to drawing.

while(brain && found==0)

{

brain.read((char\*)&p,sizeof(p));

if(querycmpi(p.query,qu)==0){

found=1;

break;

}

}//while ended

switch(found)

{

case 1:

for(i=0;p.reply[i]!='\0';i++)

{

cout<<p.reply[i];

delay(0.5\*100);

}cout<<endl;

delay(1.0\*1000);

break;

case 2:Ltheme();break;

case 3:collision(); break;

case 4:proj();break;

case 5:drawfun();break;

}

if(found==0 && (querycmpi(qu,"exit")!=0))// && querycmpi(qu,"bye")!=0))

{

brain.close();

ofstream brain2("Brain",ios::app);

cout<<"How would you reply to that ? "<<endl;

gets(repu);

if (querycmpi(repu,"-0")==0)goto GETQ;

strcpy(p.query,qu);

strcpy(p.reply,repu);

brain2.write((char\*)&p,sizeof(p));

brain2.close();

}

else if (querycmpi(qu,"exit")==0) brain.close();

for(int l=0;qu[l]!='\0';l++)query[l]=qu[l];

}while(querycmpi(query,"exit")!=0 && querycmpi(query,"bye")!=0);

cout<<"\nBye...";

return 0;

}

void Ltheme()

{

clrscr(); char ch=14;

cout<<ch<<" : L's Theme A ";

while(!(kbhit()) )

{ sound(783.99);//G3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

sound(659.26);//E3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

sound(880);//A3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

sound(659.26);//E3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

sound(739.99);//F#3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

sound(783.99);//G3

delay(0.15\*1000);

nosound();delay(25);

if(kbhit())break;

}

}

# Header file:Animatio.h

#include <stdio.h>

#include <conio.h>

#include <iostream.h>

#include <iomanip.h>

#include <process.h>

#include <string.h>

#include <math.h>

#include <dos.h>

long m;

int anim(char string[10][60], int num)

{

int i, j, l, k;

for(i=0; i<num; i++)

{

for(j=27-2\*i; j>i; j--)

{

clrscr();

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

cout<<string[k]<<endl;

for(l=j-1; l>i; l--)

cout<<endl;

cout<<string[i];

delay(10);

}

}

return 0;

}

int marquee(char string1[10][60], int no)

{

int i, j, k, l, len, length[10];

for(i=0; i<no; i++)

length[i]=strlen(string1[i]);

for(i=0; i<no; i++)

{

for(j=0; j<length[i]; j++)

{

clrscr();

len=strlen(string1[i]);

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

cout<<" ";

cout<<string1[i]<<endl;

string1[i][len-1]='\0';

for(l=i+1; l<no; l++)

cout<<string1[l]<<endl;

delay(10);

}

}

clrscr();

return 0;

}

int load()

{

int i, k=0, j;

char x=176, y=219;

for(i=0; i<79; i++)

{

delay(10);

if(i%3==0)

{ clrscr();cout<<"Loading/\n";

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

cout<<y;

for(j=0; j<79-i; j++)

cout<<x;

}

else if(i%3==1)

{

clrscr();cout<<"Loading|\n";

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

cout<<y;

for(j=0; j<79-i; j++)

cout<<x;

}

else

{ clrscr();cout<<"Loading\\\n";

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

cout<<y;

for(j=0; j<79-i; j++)

cout<<x;

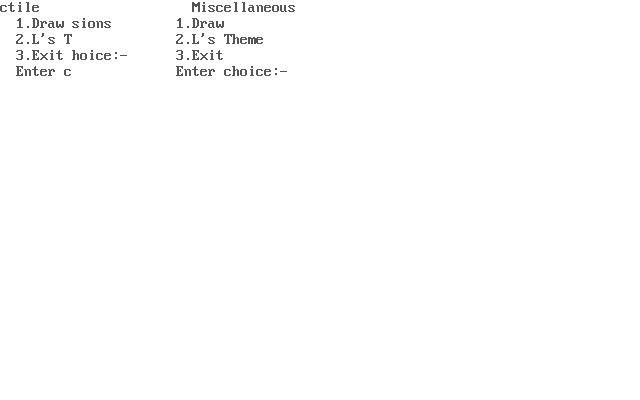
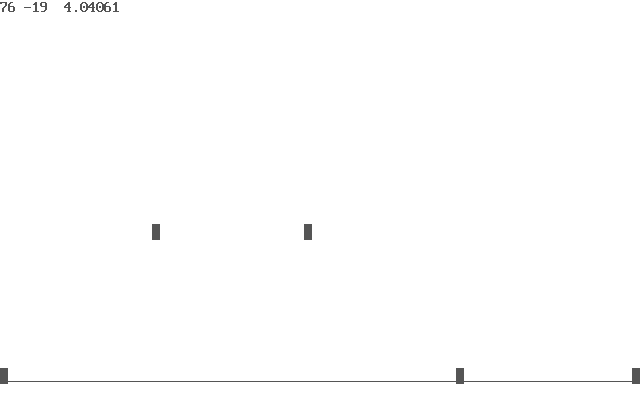
}

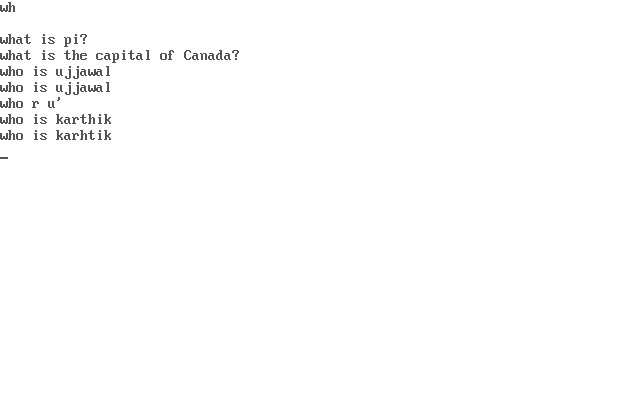
}

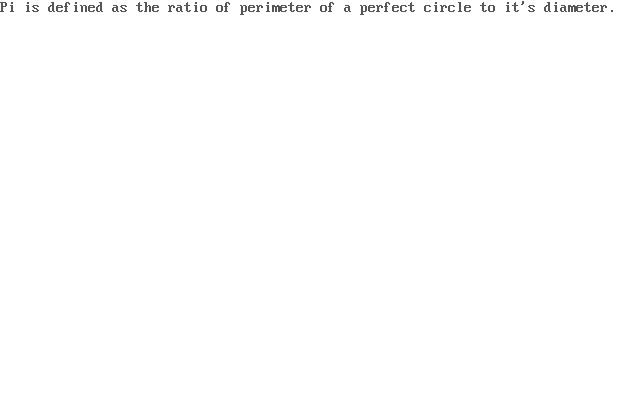
return 0;

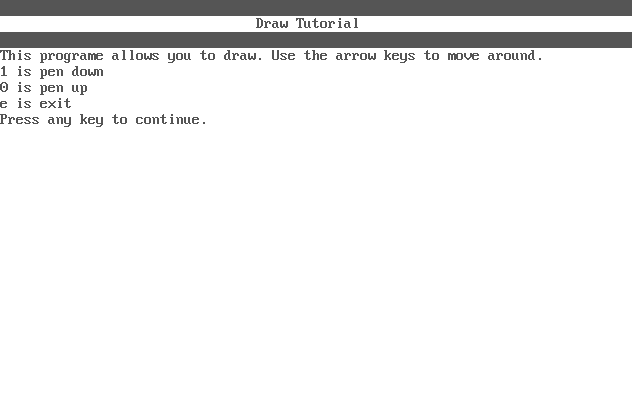
}

Images









# Acknowledgement

I would like to express my thanks of gratitude to my teacher Ms. Alpa Biju, who gave me the opportunity to do this computer project, which helped me in developing my concepts in the fields of Physics and also my logical reasoning ability, besides getting more familiar with C++.

This project is in collaboration with my classmate and good friend Unay Shah and I would like to thank him.