/\* intro.h \*/

# include <graphics.h>

# include <conio.h>

# include <dos.h>

# include <stdlib.h>

void intro()

{

clrscr();

int gd=DETECT,gm,i=0;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\tc\bgi");

rectangle(0,5,630,400);

rectangle(10,15,620,390);

settextstyle(1,0,3);

outtextxy(130,60,"SOME EYE PERCEPTION THINGY");

settextstyle(1,0,4);

outtextxy(273,120,"BY");

settextstyle(1,0,3);

outtextxy(42,178," EYE RESOLUTION SENSOR SOFTWARE (blinky)");

settextstyle(1,0,1);

outtextxy(245,238,"DESIGNER");

outtextxy(35,280,"JATIN RAJPAL ");

//outtextxy(35,310," ");

while(!kbhit())

{

setcolor(i);

settextstyle(2,0,0);

outtextxy(32,170,"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

\*

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*");

outtextxy(32,178,"\*

\*");

outtextxy(32,186,"\*

\*");

outtextxy(32,196,"\*

\*");

outtextxy(32,205,"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

\*

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*");

setfillstyle(XHATCH\_FILL,i++);

delay(200);

floodfill(2,9,15);

if(i>=15)

i=0;

}

getch();

cleardevice();

closegraph();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END OF INTRO.H\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* main file: blinky.h\*/

# include <iostream.h>

# include <conio.h>

# include <graphics.h>

# include <dos.h>

# include <time.h>

# include <stdlib.h>

# include <intro.h>

# include <backend.h>

int poly[8],toggle=1,z;

void blink(int fuzzy[],int size,int lazy,int paint)

{

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

{

randomize();

z=random(5); // random

if(!z) z=1;

poly[0]=fuzzy[i];

poly[1]=z\*fuzzy[i+1];

poly[2]=0+fuzzy[i];

poly[3]=30+z\*fuzzy[i+1];

poly[4]=30+fuzzy[i];

poly[5]=30+z\*fuzzy[i+1];

poly[6]=30+fuzzy[i];

poly[7]=0+z\*fuzzy[i+1];

if(paint==1)

paint=z;

setfillstyle(1,paint);

fillpoly(4,poly);

delay(lazy);

setfillstyle(1,0);

fillpoly(4,poly);

}

}

int roundoff(int x)

{

int temp=0;

if(x%30!=0)

{

if(x<30)

return 30;

else

{

temp=x%30;

return(30\*temp);

}

}

else return x;

}

void main()

{

int gd,gm;

detectgraph(&gd,&gm);

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

intro();

int ch,lazy,paint,size;

flag:

cout<<"

CONTROL PANEL";

cout<<"

1.Change blink rate";

cout<<"

2.Change blink color";

cout<<"

3.Specify no. of blinks per second";

cout<<"

4.Use default settings and exit";

cout<<"

5.Exit simulator";

cout<<"

Enter your choice--->";

cin>>ch;

switch(ch)

{

case 1:

cout<<"

choose type of blink rate: very slow/slow/fast/very

fast(enter

1/2/3/4)--->";

cin>>ch;

if(ch==1)

lazy=1000;

if(ch==2)

lazy=600;

if(ch==3)

lazy=100;

if(ch==4)

lazy=10;

break;

case 2:

cout<<"

blink color: random or fixed color ?(enter 1/2)--->";

cin>>paint;

break;

case 3:

cout<<"roughly estimate the no. of blinks per second(multiples of

ten,max

50)--->";

cin>>size;

break;

case 4:

lazy=100;

paint=15;

size=20;

break;

case 5:

backend();

exit(0);

}

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

for(int i=0;i<630;i+=30)

{

for(int j=0;j<470;j+=30)

{ delay(10);

rectangle(i,j,i+30,j+30);

}

}

int fuzzy[50],dummy=0;

while(!kbhit())

{

randomize();

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

{

dummy=random(480);

fuzzy[i]=roundoff(dummy);

}

dummy=random(480);

fuzzy[i+1]=roundoff(dummy);

blink(fuzzy,size,lazy,paint);

}

closegraph();

goto flag;

getch();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END OF BLINKY.H\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* back end: BACKEND.H\*/

# include <graphics.h>

# include <conio.h>

# include <dos.h>

void backend()

{

int gd=DETECT,gm;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"c:\tc\bgi");

rectangle(0,5,630,400);

rectangle(12,15,618,390);

settextstyle(1,0,5);

outtextxy(180,60,"THANK YOU");

settextstyle(1,0,3);

outtextxy(160,128,"HOPE U LIKED IT !");

settextstyle(1,0,1);

outtextxy(220,250,"HAVE A NICE DAY !");

int i=0;

while(!kbhit())

{

delay(200);

setfillstyle(XHATCH\_FILL,i++);

floodfill(2,9,15);

if(i>=15)

i=0;

}

getch();

closegraph();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END OF BACKEND.H\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/