DATE :

PROGRAM NO :1

ROTATING AN IMAGE

AIM:

Write a program to rotate an image.

ALGORITHM:

Step1: Start the program.
Step2: Initialize the graph using the initgraph( ) function.
Step3: Get the angle, the image to be rotated.
Step4: Calculate angle=angle/57.3.
Step5: Using the cos ( ) and sin ( ) function redraw the image according to the angle.
Step6: Stop the program.

PROGRAM:

```
# include <stdio.h>
# include <conio.h>
# include <graphics.h>
# include <math.h>
# define xr 320
# define yr 240
void main()
{
int x=520,y=240,xp=0,yp=0;
double ang;
int gd=DETECT,gm,errorcode;
initgraph(&gd,&gm," C:\\TURBOC3\\BGI ");
setcolor(4);
line(320,240,520,240);
printf("\nEnter the Angle to rotate : ");
scanf("%lf",&ang);
setcolor(0);
line(320,240,520,240);
setcolor(4);
ang = ang / 57.3;
xp = xr + (x-xr) * cos(-ang) - (y-yr) * sin(-ang);
yp = yr + (x-xr) * sin(-ang) - (y-yr) * cos(-ang);
line(xr,yr,xp,yp);
getch();
}
```

OUTPUT

Enter the Angle to rotate 90 :

DATE :

PROGRAM NO :2

DROPPING EACH WORD OF SENTENCE

AIM:

To write a program to drop each word of a sentence one by one from the top.

ALGORITHM:

Step 1: Start the program.
Step 2: Declare all the variables that are used in the program.
Step 3: Declare the sentence for dropping.
Step 4: Using for loop drop the word one by one.
Step 5: Display the whole sentence at the bottom of the screen after dropping.
Step 6: Stop the program.

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<stdlib.h>
#include<string.h>
#include<dos.h>
void main()
{
char *p;
char a[80];
int x=100;
int gd=DETECT,gm;
initgraph(&gd,&gm," C:\\TURBOC3\\BGI ");
clrscr();
printf("\n Enter a sentence :");
gets(a);
p=strtok(a,"");
if(p)
{
outtextxy(0,472,p);
}
while(p!='\0')
{
sleep(2);
p=strtok(NULL,"");
if(p)
{
```

```
outtextxy(x,472,p);
x=x+100;
}
}
getch();
}
```

OUTPUT:

Enter a Sentence: SREE NARAYANA GURU COLLEGE

SREE NARAYANA GURU COLLEGE

DATE :

PROGRAM NO :3

DDA LINE DRAWING ALGORITHM

AIM:

To create a program to draw a line using DDA line drawing Algorithm.

ALGORITHM:

Step1: Start the process.
Step2: Declare the variables.
Step3: Initialize the graphic system.
Step4: Give the request for auto detection.
Step5: Get the coordinate value.
Step6: Clear the screen using clear device function.
Step7: Display a text screen in the view port at a given position using Outtextxy() function.
Step8: putpixel() function is used to plot the pixel until x value is greater Than y.
Step10: Stop the process.

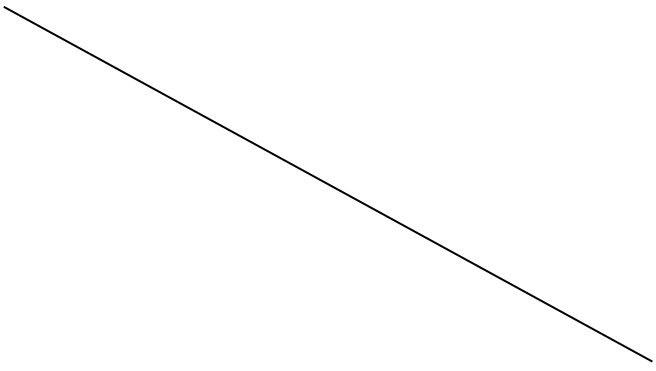
PROGRAM:

```
# include <graphics.h>
# include <conio.h>
# include <stdio.h>
# include <math.h>
int i,steps,dx,dy,xa,xb,ya,yb,d,m;
int xinc,yinc,x,y;
void main()
{
d=DETECT;
initgraph(&d,&m," C:\\TURBOC3\\BGI ");
printf("\nEnter Values X1,X2,Y1,Y2 \n");
scanf("%d %d %d %d",&xa,&xb,&ya,&yb);
outtextxy(150,150,"DDA Line Drawing Alg \n");
dx = xb - xa;
dy = yb - ya;
if(abs(dx)>abs(dy))
{
steps = abs(dx);
}
else
{
steps = abs(dy);
xinc = dx/steps;
yinc = dy /steps;
x=xa;
y=ya;
putpixel((int)x,(int)y,15);
```

```
}
for(i=1;i<=steps;i++)
{
x = x + xinc;
y = y + yinc;
putpixel((int)x,(int)y,15);
}
getch();
}
```

OUTPUT:

Enter Values X1,X2,Y1,Y2: 100 150 200 25



DATE :

PROGRAM NO :4

MOVING A CAR

AIM:

To create a program to move a car with sound effect.

ALGORITHM:

Step 1: Start the program.
Step 2: Give the request for auto detection and declare the variables.
Step 3: Initialize the graphic system.
Step 4: Color function are used to set color to next.
Step 5: Sets the current line style and width or pattern using setline style Function.
Step 6: Line function, circle function, rectangle function is used to draw line, circle, rectangle respectively.
Step 7: To set the current fill pattern and color by using setfillstyle function.
Step 8: To fill style and enclosed area into on bit map device by using flood Fill function.
Step 9: To copy an image from the screen to memory by using getimage function.
Step 10: Using putimage function move the buffer.
Step 11: Give sound effect by using sound function.
Step 12: Run the program.
Step 13: Stop the program.

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<dos.h>
void main()
{
int j,X1,X2;
int gdriver=DETECT,gmode,errorcode;
initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI");
for(j=0;j<3;j++)
{
for(X1=100,X2=150;X1<=500;X1++,X2++)
{
setcolor(4);
rectangle(X1,50,X2,75);
rectangle(X1-25,75,X2+25,100);
circle(X1,105,5);
circle(X2,105,5);
delay(10);
setcolor(0);
rectangle(X1,50,X2,75);
rectangle(X1-25,75,X2+25,100);
circle(X1,105,5);
circle(X2,105,5);
}
for(X1=500,X2=550;X1>=100;X1--,X2--)
{
```

```
setcolor(4);
rectangle(X1,50,X2,75);
rectangle(X1-25,75,X2+25,100);
circle(X1,105,5);
circle(X2,105,5);
delay(10);
setcolor(0);
rectangle(X1,50,X2,75);
rectangle(X1-25,75,X2+25,100);
circle(X1,105,5);
circle(X2,105,5);
}
}
getch();
}
```

OUTPUT:

DATE :

PROGRAM NO :5

BOUNCING THE BALL WITH SOUND EFFECT

AIM:

Write a program to bounce a ball and move it with sound effect

ALGORITHM:

Step1: Start the program.
Step2: Declare the variables that are used in the program.
Step3: Using the putimage( ) function display the ball.
Step4: Compute the value of x and y.
Step 5: Compute x=x+(x.dir*s);
Step6: Bounce the ball by changing the value.
Step7: Close the screen using closegraph( ) keyword.
Step8: Stop the program.

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<dos.h>
void main()
{
int i,j;
int gdriver=DETECT,gmode,errorcode;
initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI");
for(j=0;j<10;j++)
{ for(i=100;i<400;i++)
{
setcolor(4);
circle(100,i,10);
delay(3);
setcolor(0);
circle(100,i,10);
}
for(i=400;i>100;i--)
{
setcolor(4);
circle(100,i,10);
delay(3);
setcolor(0);
circle(100,i,10);
```

```
}
}
getch();
}
```

OUTPUT:

DATE :

PROGRAM NO :6

TESTING PIXEL IN A POLYGON

AIM:

To write a program to test whether a given pixel is inside or outside of a polygon.

ALGORITHM:

Step 1: Start the program.
Step 2: Get the co-ordinate value for the pixel position.
Step 3: Using for loop get the coordinate value of the line.
Step 4: Declare the variable that are used in program.
Step 5: Display the pixel according to the coordinate value given.
Step 6: Using if condition check whether the pixel is inside or outside or in the border of the polygon.
Step 7: Stop the program.

PROGRAM:

```
# include <stdio.h>
# include <conio.h>
# include <graphics.h>
# include <dos.h>
void main()
{
int poly[8],x,y,c;
int gdriver=DETECT,gmode,errorcode;
initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI ");
poly[0]=300;
poly[1]=200;
poly[2]=300;
poly[3]=100;
poly[4]=500;
poly[5]=200;
poly[6]=poly[0];
poly[7]=poly[1];
drawpoly(4,poly);
setcolor(11);
fillpoly(4,poly);
setcolor(4);
printf("\nEnter the Values of X & Y ");
scanf("%d %d",&x,&y);
c = getpixel(x,y);
circle(x,y,1);
if(c==15)
printf("\nPixel is in Inside");
if(c==11)
printf("\nPixel is in Border");
if(c==0)
printf("\nPixel is in Outside");
getch();
}
```

OUTPUT:

Enter the Values of X and Y 250 250

Pixel is in outside

PGM NO: 7

DATE:

CREATION OF A SUNFLOWER

AIM:

To create a sunflower using Photoshop.

ALGORITHM:

Step 1: Start the process.
Step 2: Open a new file in Photoshop.
Step 3: Create a new layer.
Step 4: Using Custom Shape Tool, draw the picture of sunflower.
Step 5: Repeat the step 4 again till the sunflower shape.
Step 6: Rotate sunflower using move tool while right click on the image for Free Transform (or) press ctrl + T.

Step 7: draw a stem and leaf using brush tool or shape tool (line and leaf) with green color. using Filter-> Liquify option to bend the stem.

Step 8: Rotate shape of a leaf using move tool while right click on the image for Free Transform (or) press ctrl + T and rotate leaf according to stem.

Step 9: fill the inner part of the flower with brown color using custom shape or using brush. Save it with image file format.

Step 10: Stop the process

OUTPUT:

SUNFLOWER

PGM NO:8

DATE:

FLYING AN AEROPLANE

AIM:

To animate the plane flying in the clouds.

ALGORITHM:

Step1: Start the process.
Step2: Open a new file in Photoshop.
Step3: Create a background with different clouds.
Step4: Create new layer and draw new clouds.
Step5: Repeat Step 3 twice.
Step6: Create new layer(layer -> new layer) and draw aeroplane by using custom shape tool.
Step7: Apply suitable color for the plane using color palette.
Step8: Click jump to image ready & select window -> animation for a plane.
Step9: Insert a duplicate frame from the animation palette.
Step10: Change the visibility of each frame from the document window.
Step11: Set the duration of the plane during play fade. Click play button from the animation palette.
Step12: Save and close the file.

.

OUTPUT:

FLYING AN AEROPLANE

PGM NO:9

DATE:

PLASTIC SURGERY FOR NOSE

AIM:

To create plastic surgery for the nose.

ALGORITHM:

Step 1: Open a new file.
Step 2: Open an image.
Step 3: take duplicate copy of the image using Image-> Duplicate.
Step 4: Select nose using square selection tool.
Step 5: i) Press ctrl+T and adjust the shape of nose Using “Clone Stamp Tool” apply some adjustment. Press right click on image and click deselect option to deselect the nose.

or

ii) Choose “Filter-> Liquify” and apply adjustment, click ok.
Step 8: Save and close.

NOSE SURGERY

Before surgery

After surgery

PGM NO:10

DATE:

SEE-THROUGH TEXT

AIM:

To create see-through text.

ALGORITHM:

Step 1: Open New document File->New.
Step 2: Open any photo File->Open and take duplicate picture (image -> Duplicate).
Step 3: Click Horizontal type mask tool and type.
Step 4: Select typed character and select Size, Font, color and click Move tool.
Step 5: Click Edit->Copy merged.
Step 6: Click Edit->Paste, Then we can see the characters through the image.
Step 7: Save and close

SEE THROUGH TEXT

PGM NO: 11

DATE:

CREATION OF A WEB PAGE

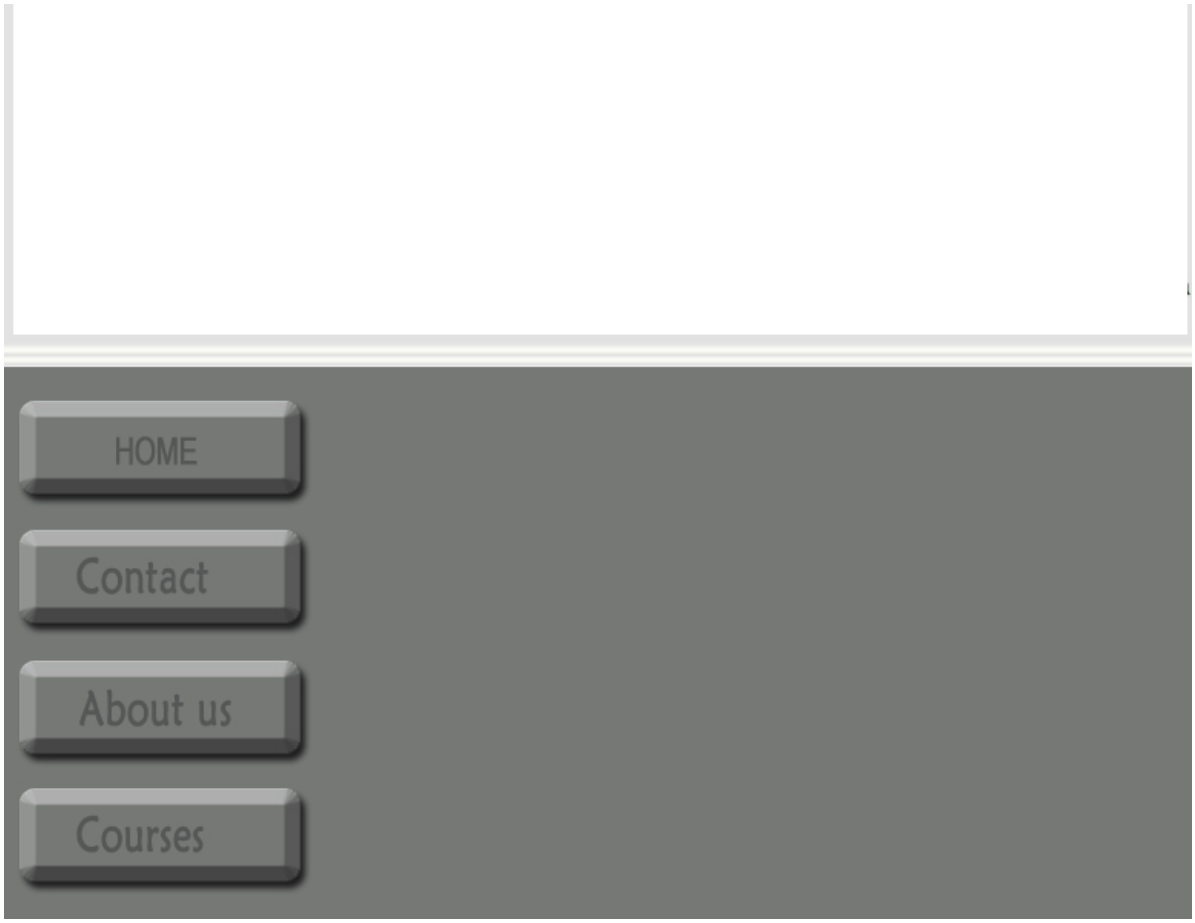
AIM:

To create a web page.

ALGORITHM:

Step 1: Start the process.
Step 2: open a adobe Photoshop and click image ready icon in the tool box
Step 3: open new document.
Step4: Using slice tool, create a box as entire slice web page.
Step 4: In top of the webpage can be designed using textbox.
Step 5: In bottom of the webpage, hyperlink can be created using Rectangle image map tool, image can also placed with in the image map and text can be inserted using textbox.
Step 6: html pages can be created according to the web page creation.
Step 7: links are created by window->image map-> to assign URL, Target of the web page.
Step 8: Create another image by repeating page step 5 to 7.
Step 9: to execute the webpage using internet explorer icon in tool box
Step 10: Stop the process.

WEB PAGE CREATION



PGM NO: 12

DATE:

BLACK AND WHITE IMAGE TO COLOUR

AIM:

- To convert a black and white image to a colour one.

ALGORITHM:

Step1: Open an already existing between image to File->open an image to To black and white format. Take duplicate image using image-> duplicate.
Step to be followed are:

a) Open the colored image ->image menu->Gray scale->Click yes or press ctrl+1 to change the image into black and white .

b) Image->mode->RGB color->Click ok

Step2: Now create duplicate layer by clicking Layer->Duplicate layer->Ok.

Step3: Select a separate portion of the image using magnetic lasso tool. The procedure to be followed is:
Sub step1: select the paint bucket tool and color then apply to the selected portion of an image.

Step4: You make this portion colour Click image->adjustment->colour Balance in the window (Ctrl+U). Then change the level of cyan, magenta,Yellow. Finally click ok.

Step5: Press Ctrl+t for deselect icon and repeat the above steps to all the other Parts

BLACK & WHITE TO COLOR