

1/1/out
M T W T F S S

COMPASS
Date:

1. Program to Do DDA Line Diagram. Algorithm

```
#include <iostream.h>
#include <graphics.h>
#include <conio.h>
#include <math.h>
void main()
{
    int gd=DETECT, gm;
    initgraph(&gd,&gm,"..\\BGI");
    float x1, y1, x2, y2, dx, dy, xinc, yinc, i, steps;
    cout << "In enter starting points ";
    cin >> x1 >> y1;
    cout << "In enter end points ";
    cin >> x2 >> y2;
    dx = x2 - x1;
    dy = y2 - y1;
    if (dx > dy)
        steps = dx;
    else
        steps = dy;
    xinc = dx / steps;
    yinc = dy / steps;
    for (i=0; i<=steps; i++)
    {
        putpixel(x1, y1, 2);
        x1 = x1 + xinc;
        y1 = y1 + yinc;
        getch();
        closegraph();
    }
}
```

M	T	W	T	F	S	S
<input type="checkbox"/>						

O/p

Enter starting point 100 200

Enter end points 200 300

1/01/22

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMPASS

Date :

22.

2

program to do BRC SENTHAN's line drawing
algorithm

```
#include <iostream.h>
#include <conio.h>
#include <graphics.h>
Void main ()
{
    int gd = DETECT, gm;
    float x1, x2, y1, y2, x, y, d, dy, dx;
    initgraph (&gd, &gm, "..\\BGI");
    cout << "enter co-ordinates of line";
    cin >> x1 >> y1 >> x2 >> y2;
    dx = x2 - x1;
    dy = y2 - y1;
    x = x1;
    y = y1;
    d = d + dy - dx;
    while (x < x2)
    {
        putpixel (x, y);
        if (d >= 0)
        {
            x++;
            y++;
            d = d + 2 * (dy - dx);
        }
        else
        {
            x++;
            d = d + 2 * dy;
        }
    }
}
```

Algorithm.

of the line
start

and dx

M	T	W	T	F	S	S
<input type="checkbox"/>						

getch();
closegraph();
?

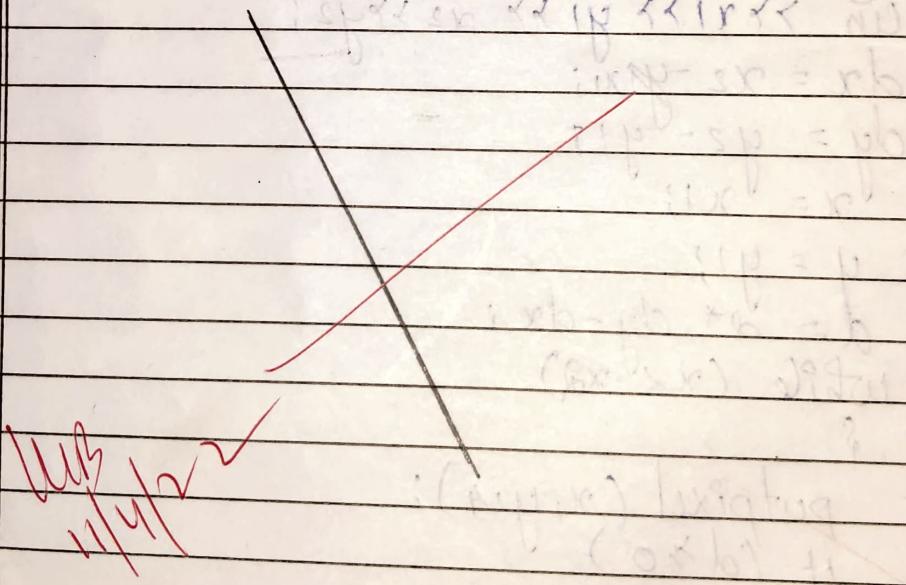
output

enter coordinate of line 100

200

300

400



18/04/22
25/04/22

M T W T F S S

COMPASS

Date:

③ program To do DDA circle Drawing Algorithm.

```
#include <iostream.h>
#include <conio.h>
#include <graphics.h>
#include <math.h>
void main()
{
    int gd=DETECT,gm;
    float xc,yc,x,y,t,s,r;
    initgraph(&gd,&gm,"..\\BGI");
    cout << "enter the coordinate of center";
    cin >> xc >> yc;
    cout << "enter radius";
    cin >> r;
    for(t=0;t<=(2*3.14);t+=0.01)
    {
        x= xc + r * cos(t);
        y= yc + r * sin(t);
        putpixel(x,y,3);
    }
    getch();
    closegraph();
}
```

M T W T F S S

COMPASS

Date:

output

Enter the coordinates of circle 200
200

enter radius 50



25/04/22

M T W T F S S

COMPASS

Date:

① Program to do Bresenham's circle drawing algorithm

```
#include <iostream.h>
#include <conio.h>
#include <graphics.h>
```

```
void plot(int a, int b, int xc, int yc)
```

```
{  
    putpixel(xc+a, yc+b, 1);  
    putpixel(xc+b, yc+a, 2);  
    putpixel(xc-a, yc+b, 3);  
    putpixel(xc+a, yc-b, 4);  
    putpixel(xc-b, yc+a, 5);  
    putpixel(xc+b, yc-a, 6);  
    putpixel(xc-a, yc-b, 7);  
    putpixel(xc-b, yc-a, 8);  
}
```

```
void main()
```

```
{  
    int gd=DETECT, gm;  
    float xc, yc, r, p, x, y;  
    initgraph(&gd, &gm, "..\\BGI");  
    cout << "Enter the coordinates and radius";  
    cin >> xc >> yc >> r;  
    x=0, y=r;  
    p=1-r;
```

```
    while (x < y)  
    {
```

```
        if (p < 0)
```

~~x++;~~

```
        plot(x, y, xc, yc);
```

M	T	W	T	F	S	S

```
? p = p + 2*x + 1;  
else  
{  
    x += 1, y -= 1;  
    plot(x, y, xc, yc);  
    p = p + 2*(x - y) + 1;  
}  
getch();  
closegraph();
```

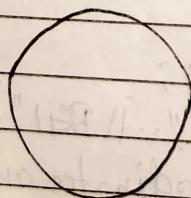
output

Enter the coordinates and radius

200

200

50



18/04/16

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMPASS

Date :

5 Study the basic functions defined in "graphics.h"

```
#include <graphics.h>
```

```
#include <conio.h>
```

```
Void main
```

```
{
```

```
int gd = DETECT, gm;
```

```
initgraph(&gd, &gm, "C:\BGI");
```

```
Circle(100, 100, 50);
```

```
ellipse(100, 200, 0, 360, 50, 25);
```

```
rectangle(300, 275, 100, 050);
```

```
bar(500, 10, 10, 10);
```

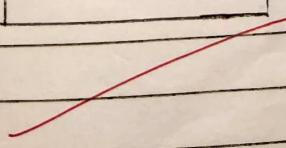
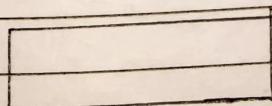
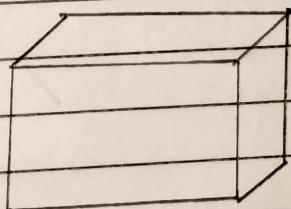
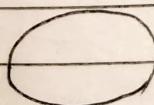
```
boxed(360, 255, 200, 200, 20, 1);
```

```
getch();
```

```
closegraph();
```

```
}
```

Output



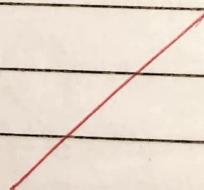
WAP

city

M T W T F S S
X/ / / / / /

6. Write a program to draw a hut or a geometric figure

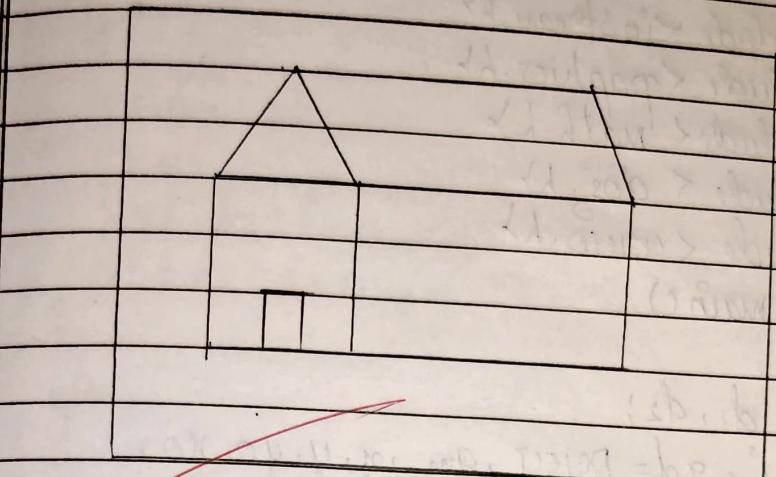
```
#include <iostream.h>
#include <conio.h>
#include <graphics.h>
#
void main()
{
    gd=DETECT,gm;
    initgraph(&gd,&gm,"..\\BGI");
    line(150,100,50,200);
    line(150,100,350,100);
    line(150,100,300,200);
    line(300,200,500,200);
    line(350,100,500,200);
    line(50,200,300,200);
    rectangle(50,40,300,200);
    rectangle(300,200,500,400);
    rectangle(130,250,230,400);
    getch();
}
```



M T W T F S S

COMPASS
Date:

output



Up
25/4



09/05

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMPASS

Date:

#

Program to do Ellipse Generation algorithm

```

#include <iostream.h>
#include <graphics.h>
#include <math.h>
#include <dos.h>
#include <conio.h>
Void main()
{
    long d1, d2;
    int i, gd = DETECT, gm, rx, ry, xo, yo;
    long rx2, ry2, rxsg, rysg, tuorxsg, tuorysg, dx, dy;
    clrscr();
    Putgraph(&gd, &gm, "..\\BGI");
    cout << "Enter the x radius and y radius of the ellipse: \n";
    cin >> rx >> ry;
    cleardevice();
    cout << "Enter the center(x,y) of the ellipse: \n";
    cin >> xo >> yo;
    rxsg = rx * rx;
    rysg = ry * ry;
    tuorxsg = 2 * rxsg;
    tuorysg = 2 * rysg;
    x = 0;
    y = ry;
    d1 = rysg - rxsg * ry + (0.25 * rxsg);
    dx = tuorxsg * x;
    dy = tuorysg * y;
    do
    {
        putpixel(xo + x, yo + y, 15);
        putpixel(xo - x, yo - y, 15);
        if (d1 < 0)
            d1 = d1 + 2 * rxsg * y + 1;
        else
            d1 = d1 + 2 * rxsg * y - 2 * rysg + 1;
        x++;
        y--;
    } while (x <= rx);
}

```

09/05

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMPASS
Date:

putpixel(x0+x, y0-y, 15);
 putpixel(x0-x, y0+y, 15);
 if (d1 < 0)
 {

x = x + 1;

y = y;

~~dx = dx + fuorxysq;~~

d1 = d1 + dx + dy sq;

}

else

{

x = x + 1;

y = y - 1;

~~dx = dx + fuorxysq;~~

~~dy = dy - fuorxysq;~~

~~d1 = d1 + dx - dy + dy sq;~~

}

}

while (dx < dy),

$d2 = rysq + (x+0.5) * (x+0.5) + rxsq * (y-1) * (y-1) - (rxsq * dy sq);$

do

{

putpixel(x0+x, y0+y, 15);

putpixel(x0-x, y0-y, 15);

putpixel(x0+x, y0-y, 15);

putpixel(x0-x, y0+y, 15);

if (d2 > 0)

{

x = x;

y = y - 1;

$dy = dy - fuorxysq;$

M T W T F S S

COMPASS
Date:

$$d_2 = d_2 - dy + rx \cdot sq;$$

?

else

{

$$x = x + 1;$$

$$y = y - 1;$$

$$dx = dx + fuor \cdot y \cdot sq;$$

$$dy = dy - fuor \cdot x \cdot sq;$$

$$d_2 = d_2 + dx - dy + rx \cdot sq;$$

?

while ($y > 0$);

getch();

closegraph();

?

8/10/5

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMBASS
Date:

7. write a program to translate a triangle according to user choice.
8. write a program to draw a line and increase its length three times
9. MAP draw a triangle and scale it according to user choice

```
#include <iostream.h>
#include <graphics.h>
#include <conio.h>

Void translate (int *x1, int *y1, int *x2, int *y2,
                int *x3, int *y3, int *tx, int *ty)
{
    *x1 = *x1 + *tx;
    *x2 = *x2 + *tx;
    *x3 = *x3 + *tx;
    *y1 = *y1 + *ty;
    *y2 = *y2 + *ty;
    *y3 = *y3 + *ty;
}
```

```
void main ()
```

```
int gd = DETECT, gm;
initgraph (&gd, &gm, "..\BGI");
int x1 = 100, y1 = 100, x2 = 150, y2 = 50, x3 = 200,
          y3 = 100, tx, ty;
```

cout << "enter the translation factor";

cin >> tx >> ty;

line (x1, y1, x2, y2);

line (x2, y2, x3, y3);

M	T	W	T	F	S	S
<input type="checkbox"/>						

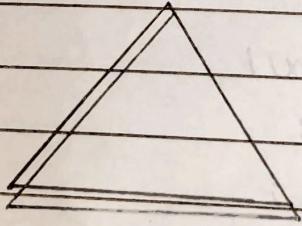
line (x_3, y_3, x_1, y_1);
translate (& $x_1, &y_1, &x_2, &y_2, &x_3, &y_3,$
 $\&tx, \&ty$);
line (x_1, y_1, x_2, y_2);
line (x_2, y_2, x_3, y_3);
line (x_3, y_3, x_1, y_1);
getch();
closegraph();

2

Q8??

enter the translation factor 2

3



09/05

M	T	W	T	F	S	S
<input type="checkbox"/>						

COMPASS

Date:

9

```
#include <iostream.h>
#include <graphics.h>
#include <conio.h>
void Scaling (int *x1, int *y1, int *x2, int *y2,
              int *x3, int *y3, int *ax, int *ay)
{
    *x1 = *x1 * *ax;
    *x2 = *x2 * *ax;
    *x3 = *x3 * *ax;
    *y1 = *y1 * *ay;
    *y2 = *y2 * *ay;
    *y3 = *y3 * *ay;
}
```

?

Void main ()

{

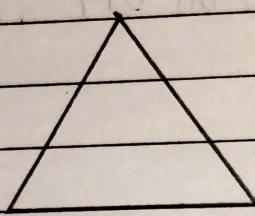
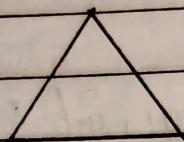
```
int gd = DETECT, gm;
initgraph (&gd, &gm, "..\\BGI\\");
int x1 = 100, y1 = 100, x2 = 150, y2 = 50, x3 = 200,
        y3 = 100, ax, ay;
cout << "enter the scaling factor";
cin >> ax >> ay;
line (x1, y1, x2, y2);
line (x2, y2, x3, y3);
line (x3, y3, x1, y1);
Scaling (&x1, &y1, &x2, &y2, &x3, &y3, &ax, &ay);
line (x1, y1, x2, y2);
line (x2, y2, x3, y3);
line (x3, y3, x1, y1);
getch();
closegraph();
```

W/B/als

M	T	W	T	F	S	S
<input type="checkbox"/>						

Enter the Scaling Factor \times

3



M T W T F S S

COMPASS

Date:

10. WIAPI to draw a Square and rotate it by 45 degree

11. WIAPI menu driven Program to draw a triangle and reflect it along the following axis

- along x-axis
- Along y-axis
- Along x=y axis

10. #include <iostream.h>

#include <conio.h>

#include <graphics.h>

Void main()

{

int gd= DETECT ; gm;

initgraph (&gd, &gm, ".\BG1");

float x1, y1, x2, y2, x3, y3, x4, y4, x5, y5, x6, y6,
x7, y7, x8, y8, rad;

int arr[] = {x1, y1, x2, y2, x3, y3, x4, y4, x5, y5, x6, y6};

cout << "Enter coordinates of the square:";

cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3 >> x4 >> y4 >> x5 >> y5;

int arr[] = {x1, y1, x2, y2, x3, y3, x4, y4, x5, y5};

drawpoly (arr, order);

rad= PI/4;

x5= x1 * cos(rad) - y1 * sin(rad);

y5= x1 * sin(rad) + y1 * cos(rad);

x6= x2 * cos(rad) - y2 * sin(rad);

y6= x2 * sin(rad) + y2 * cos(rad);

x7= x3 * cos(rad) - y3 * sin(rad);

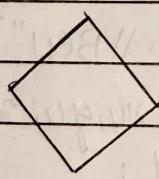
y7= x3 * sin(rad) + y3 * cos(rad);

M	T	W	T	F	S	S
<input type="checkbox"/>						

```
x8 = x4 * cos(rad) - y4 * sin(rad);  
y8 = x4 * sin(rad) + y4 * cos(rad);  
int ax1() = {x5, y5, x6, y6, x7, y7, x8, y8, x5, y5};  
drawpoly(S, ax1());  
setcolor(RED);  
closegraph();  
q
```

Output

Enter the coordinates of the Square 100 100 100
150 150 150 150 100



M T W T F S S

COMPASS
Date:

⑪
#include <iostream.h>
#include <conio.h>
#include <graphics.h>

int xm, ym, x1, y1, x2, y2, x3, y3;
Void draw()

{

line (xm+x1, ym+y1, xm+x2, ym+y2);
line (xm+x2, ym+y2, xm+x3, ym+y3);
line (xm+x3, ym+y3, xm+x1, ym+y1);

}

Void main()

{

int gd=DETECT, gm;

initgraph (&gd, "gm", ".\11BG1");

int n;

xm= gd maxx() / 2;

ym= getmaxy() / 2;

cout << "Enter the 6 values \n";

cin >> n >> y1 >> x2 >> y2 >> x3 >> y3;

draw();

cout << "1. xaxis \n";

cout << "2. yaxis \n";

cout << "3. origin \n";

cin >> n;

getch();

Switch (n);

{

case 1:

y1 = -y1;

y2 = -y2;

y3 = -y3;

M	T	W	T	F	S	S
<input type="checkbox"/>						

draw();

break;

Case 2:

$x_1 = -x_1;$

$x_2 = -x_2;$

$x_3 = -x_3;$

draw();

break;

Case 3:

$x_1 = -x_1;$

$x_2 = -x_2;$

$x_3 = -x_3;$

$y_1 = -y_1;$

$y_2 = -y_2;$

$y_3 = -y_3;$

draw();

break;

?

getch();

?

Output

Enter the 6 values:

100 150 200 150 150 100

1. X-axis

2. Y-axis

3. origin

M T W T F S S

COMPASS

Date:

12

write a program to draw a Square and
Shear it along x axis and y axis

#include <iostream.h>
#include <conio.h>
#include <graphics.h>

Void main()

int gd = DETECT, gm, x, y, x1, y1, x2, y2, y3, x3, s;
initgraph (&gd, &gm, "C:\BGI");

cout << "Enter the coordinates:";

cin >> x >> y >> x1 >> y1 >> x2 >> y2 >> x3 >> y3 >> s;

cout << "Enter the Shearing factor:";

(in %);

line (x, y, x1, y1);

line (x1, y1, x2, y2);

line (x2, y2, x3, y3);

line (x3, y3, x, y);

x = x + y * s;

x1 = x1 + y1 * s;

x2 = x2 + y2 * s;

x3 = x3 + y3 * s;

line (x, y, x1, y1);

line (x1, y1, x2, y2); Enter the coordinate of the Square

line (x2, y2, x1, y1);

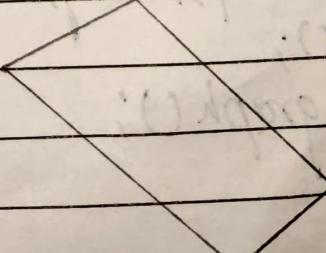
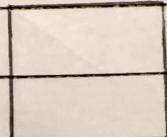
getch();

closegraph();

Enter the Shearing factor

along x axis - z

?



M	T	W	T	F	S	S
<input type="checkbox"/>						

(B)

```
#include <iostream.h>
#include <conio.h>
#include <graphics.h>
```

void main()

{

int gd = DETECT, gm; int y, x1, y1, x2, y2, x3, y3, s;

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

cout << "enter the coordinates";

cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3;

cout << "enter the shearing factor";

cin >> s;

line (x, y, x1, y1);

line (x1, y1, x2, y2);

line (x2, y2, x3, y3);

line (x3, y3, x, y);

$$y = y + x * s;$$

$$y_1 = y_1 + x_1 * s;$$

$$y_2 = y_2 + x_2 * s;$$

$$y_3 = y_3 + x_3 * s;$$

line (x, y, x1, y1);

line (x1, y1, x2, y2);

line (x2, y2, x3, y3);

line (x3, y3, x, y);

Enter the coordinates of

the square: 100 100 100

150 150 150 150 100

Enter the shearing factor

along y-axis:

line (x, y, x1, y1);

line (x1, y1, x2, y2);

line (x2, y2, x3, y3);

line (x3, y3, x, y);

getch();

closegraph();



M T W T F S S

COMPASS

Date:

(4)
(5)

ways to perform composite translation
COPAP to perform composite scaling.

(14)

```
#include <iostream.h>
#include <graphics.h>
#include <conio.h>

void trans(int *x1, int *y1, int *x2, int *y2,
           int *x3, int *y3, int *tx1, int *ty1, int *tx2,
           int *ty2)

{
    *x1 = *x1 + *tx1 + *tx2;
    *x2 = *x2 + *tx1 + *tx2;
    *x3 = *x3 + *tx1 + *tx2;
    *y1 = *y1 + *ty1 + *ty2;
    *y2 = *y2 + *ty1 + *ty2;
    *y3 = *y3 + *ty1 + *ty2;
}

void main()
{
    int gd=DETECT, gm;
    Init graph(&gd, &gm, "..\\"BG1");
    int x1=100, y1=100, x2=150, y2=50,
        x3=200, y3=100, tx1, ty1, tx2, ty2;
    cout << "enter the translation factor";
    cin >> tx1 >> ty1 >> tx2 >> ty2;
    line (x1, y1, x2, y2);
    line (x2, y2, x3, y3);
    line (x3, y3, x1, y1);
    trans (&x1, &y1, &x2, &y2, &x3, &y3, &tx1, &ty1,
           &tx2, &ty2);
}
```

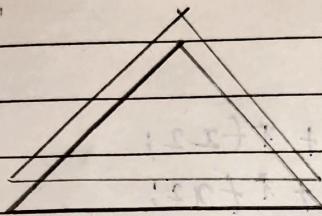
M	T	W	T	F	S	S
<input type="checkbox"/>						

```

line (x1, y1, x2, y2);
line (x2, y2, x3, y3);
line (x3, y3, x1, y1);
 getch();
 closegraph();
?
```

output:

Enter the translation factor 2 5 3 8



(15) #include <iostream.h>

#include <conio.h>

#include <graphics.h>

#include <stdlib.h>

using namespace std;

{

int gd=DETECT, gm; int x, y, r;

initgraph(&gd, &gm, ".\|BGI\|");

x=getmaxx()/2;

y=getmaxy()/2;

r=0;

setcolor(RED);

~~if(kbhit() != 'q')~~

{

circle(x,y,r);

r=r+2;

getch();

M T W T F S S

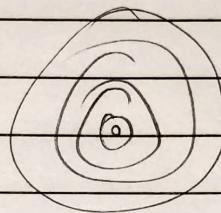
COMPASS
Date:

if ($x+r = \text{getmaxx}()$ || $y+r = \text{getmaxy}()$)
break;

?

getchar;

Output



/

Anton
8/6/22