Arc Formula:
arc(x,y,startingAngle,endingAngle,radius); Rotation Anticlockwise
arc(500,500,0,180,100);

Circle Formula:
circle(x,y,radius);

Line Formula:
line(x1,y1,x2,y2);

Out Text XY:
outtextxy(x,y,"Text");

Color:
setcolor(GREEN); //Color Lower Lines/texts

Rotation: *****
NB: value double nite hbe and degrere to radian
radian = (angle*3.14159) / 180.0;

x' = xcosA - ysinA
y' = xsinA + ycosA

Hence,
x' = x1 + (x2 - x1)*cosA - (y2 - y1)*sinA
y' = y1 + (x2 - x1)*sinA + (y2 - y1)*cosA

DDA Algorithm:

```
void dda(int xa,int ya,int xb, int yb){
    int dx=xb-xa, dy=yb-ya, steps;
    float x=xa,y=ya , xi,yi;

    if(abs(dx) > abs(dy)) steps=abs(dx);
    else steps=abs(dy);

    xi=dx/(float)steps;
    yi=dy/(float)steps;
    putpixel(ROUND(x),ROUND(y),RED);

    for(int k=0;k<steps;k++){
        x+=xi;
        y+=yi;
        putpixel(ROUND(x),ROUND(y),RED);
        delay(100);
```

```
    }
}
```

Bresenham Line Drawing :

```
void bresenham(int xa,int ya, int xb, int yb){
    int dx,dy,twoDy,twoDyDx,p,x,y,xEnd;

    dx=xb-xa,  dy=yb-ya, p=2*dy - dx,  twoDy=2*dy, twoDyDx=2*(dy-dx);

    if(xa>xb){
        x=xb;
        y=yb;
        xEnd=xa;
    }else{
        x=xa;
        y=ya;
        xEnd=xb;
    }
    putpixel(x,y,RED);
    while(x<xEnd){
        x++;
        if(p<0){
            p+=twoDy;
        }else{
            y++;
            p+=twoDyDx;
        }
        putpixel(x,y,RED);
        delay(100);
    }

}
```

MidPoint Circle Drawing Algorithm:

```
void circleDrawing(int xc,int yc,int x,int y){
    putpixel(xc+x,yc+y,WHITE);
    putpixel(xc-x,yc+y,WHITE);
    putpixel(xc+x,yc-y,WHITE);
    putpixel(xc-x,yc-y,WHITE);

    putpixel(xc+y,yc+x,WHITE);
    putpixel(xc-y,yc+x,WHITE);
    putpixel(xc+y,yc-x,WHITE);
    putpixel(xc-y,yc-x,WHITE);

    delay(100);
```

```
}
void midpoint(int xc,int yc, int r){
    int x=0, y=r;
    int p=1-r;
    void circleDrawing(int, int, int , int);
    circleDrawing(xc,yc,x,y);
    while(x<y){
        x++;
        if(p<0){
            p+=2*x +1;
        }else{
            y--;
            p+=2*(x-y)+1;
        }
        circleDrawing(xc,yc,x,y);
        delay(100);
    }
}
```