**实验一 图形基元的绘制**

**一、实验目的**

通过编程理解一些基本图形的绘制方法，加深对计算机图形显示学的理解，掌握基本图形的绘制方法。

熟悉绘制基本图形的常用算法，采用MATLAB语言实现对直线及圆用不同的算法进行绘制，加深对算法的理解。

**二、实验内容及要求**

**实验要求：**要求采用MATLAB语言进行编程，不可直接调用画直线和圆的函数。

**实验内容：**绘制以(1,2)，(7,6)为端点的直线，不可直接调用MATLAB中画直线的函数。绘制以r**=**4为半径，(8,8)为圆心的圆，不可直接调用MATLAB中画圆的函数。

**三、MATLAB编程基础**

MATLAB中程序结构流程的控制

① 顺序结构

②.循环结构

1. for -end ：特点是循环判断条件通常是对循环次数的判断，即循环次数是预先设定的。调用格式为：

                for 循环变量=起始值：步长：终止值  
                      执行语句  
                      ……  
                      执行语句  
                end  
    举例：计算1+2+3+…+100

clear

sum=0

for i=1:1:100

sum=sum+i;

end

sum  
2. while-end：判断控制一般是逻辑判断语句，使用范围更大。调用格式为：

                while 表达式  
                      执行语句  
                      ……  
                      执行语句   
                  end  
表达式一般是关系运算式或逻辑运算式组成的逻辑判断语句，以确定循环是否继续。通常表达式的值非零，即为逻辑真，程序继续循环，否则停止循环。

举例：

clear

sum=0;i=0;

while(i<=100)

sum=sum+i

i=i+1;

end

 ③. 选择结构

if-else-end 又称为条件语句，根据表达式的情况判断是否满足条件来确定程序下一步的运行。大致可分为三步进行：  
      首先计算紧跟if后面表达式。  
      接着判断表达式计算结果，若结果为0，判断值为假；若结果为1，判断值为真。  
      然后若判断值为真，则执行其后的执行语句组；否则跳过，执行下一个条件表达式或者结束该选择语句。  
  
调用格式为：

1. if 表达式  
       执行语句组  
   end

2. if 表达式  
       执行语句组1  
  else

     执行语句组2  
   end  
  
3. If 表达式1  
        执行语句组1  
    elseif 表达式2  
执行语句组2  
    elseif 表达式3  
        执行语句组3  
        ……  
   else   
        执行语句组n  
    end  
  
    关键字if 或elseif后面的条件表达式为条件，通常是由关系运算或与逻辑运算式组成的逻辑判断语句，如果if 或elseif后面的表达式的值为真时，执行紧跟其后的语句内容，否则跳过去，并根据选择语句的表达形式执行后面的elseif表达式语句、跟在else后的执行语句或end语句。

举例：中点画线法

function midpointline(x0,y0,x1,y1)

x=x0;y=y0;

b=x1-x0;a=y0-y1;

grid on;hold on;

plot(x,y,'o');

d0=a+2\*b;

d1=2\*a;

d2=2\*(a+b);

while (x<x1)

if(d0 >=0) %取下点，x+1，y不变，d的增量为2a

x=x+1;

d0 =d0+d1;

elseif(d0 < 0)%取上点，x+1，y+1，d的增量为2a+2b

x=x+1;y=y+1;d0=d0+d2;

end

hold on;

plot(x,y,'o');

end

end

④. 分支语句

switch-case-end 又称为开关语句，使程序在不同的情况下进行相应的操作。调用格式为：

   switch 表达式  
          case 常量表达式1  
           执行语句组1  
          case 常量表达式2  
           执行语句组2  
            ……  
          case 常量表达式n  
           执行语句组n  
          otherwise   
            执行语句组n+1  
    end

在switch后面的表达式为开关条件，它可以是数字或字符串。当表达式的值与某个case后面的常量表达式的值相等时，就执行相应的语句组，若没有值与所有常量表达式的值相等，执行otherwise后面的语句组。与C语言的switch不同的是，在MATLAB中，当程序执行完某个case语句组后便会立即直接跳出switch语句，执行后续的语句。

**四、实验结果**

直线绘制的MATLAB代码：

%file：drawLine.m

% 电子信息工程2017-4 #21 王新钰

% 2020年3月10日14点35分

% lab1:绘制以(1,2)，(7,6)为端点的直线，不可直接调用MATLAB中画直线的函数。

% 本程序调用matlab自带函数仅用于对比

close all

clear

%方法1：中点画线法

figure(1)

midpointLine(1,2,7,6)

%方法2：DDA画线法

figure(2)

DDALine(1,2,7,6)

%方法3：Bresenham画线法

figure(3)

BresenhamLine(1,2,7,6)

function midpointLine(x0,y0,x1,y1)

    %重点画线法函数

    x = x0;

    y = y0;

    plot([x0 x1],[y0 y1],'b-');

    axis equal;

    title('中点画线法');

    b = x1 - x0;

    a = y0 - y1;

    grid on,hold on;

    plot(x,y,'o');

    d0 = a + 2\*b;

    d1 = 2 \* a;

    d2 = 2 \* (a+b);

    while x<x1

        if d0>=0  %取下点，x+1，y不变，d的增量为2a

            x = x+1;

            d0 = d0+d1;

        elseif d0<0%取上点，x+1，y+1，d的增量为2a+2b

            x = x+1;

            y = y+1;

            d0 = d0+d2;

        end

        hold on;

        plot(x,y,'ro','MarkerSize',5,'MarkerFaceColor','g');

    end

end

function DDALine(x0,y0,x1,y1)

    %DDA画线法函数

    hold on; axis equal;

    grid on;

    plot([x0 x1],[y0 y1],'b-');

    title('DDA画线法');

    if abs(x1-x0) >= abs(y1-y0)

        length = abs(x1-x0);

    else

        length = abs(y1-y0);

    end

    dx = (x1-x0)/length;

    dy = (y1-y0)/length;

    x = x0;

    y = y0;

    i = 1;

    while i <= length

        plot(round(x),round(y),'ro','MarkerSize',5,'MarkerFaceColor','g');

        x = x + dx; y = y + dy;

        i = i + 1;

    end

    plot(round(x1),round(y1),'ro','MarkerSize',5,'MarkerFaceColor','g');

    hold off;

end

function bresenhamLine(x0,y0,x1,y1)

hold on; axis equal;

grid on;

plot([x0 x1],[y0 y1],'b-');

title('Bresenham画线法');

x = x0; y = y0;

dx = abs(x1-x0);

dy = abs(y1- y0);

s1 = sign(x1-x0);

s2 = sign(y1-y0);

if dy > dx

    temp = dx;

    dx = dy;

    dy = temp;

    interchange = 1;

else

    interchange = 0;

end

e = 2\*dy-dx;

for i = 1:dx

    plot(x,y,'ro','MarkerSize',5,'MarkerFaceColor','g');

    if e > 0

        if interchange == 1

            x = x+s1;

        else

            y = y+s2;

        end

        e = e-2\*dx;

    end

    if interchange == 1

        y = y+s2;

    else

        x = x+s1;

    end

    e = e+2\*dy;

end

plot(x1,y1,'ro','MarkerSize',5,'MarkerFaceColor','g');

hold off;

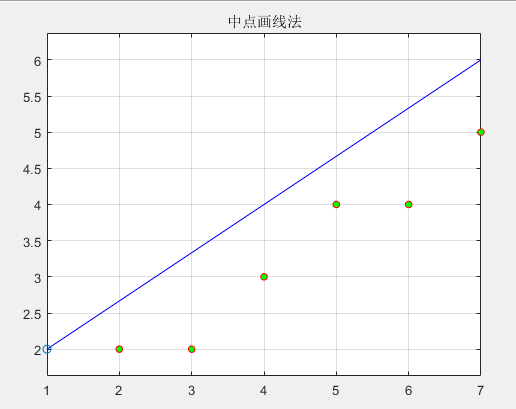


Figure 1 中点画线法

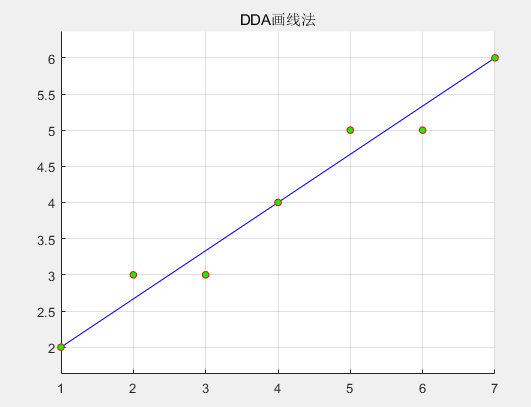


Figure 2 DDA画线法

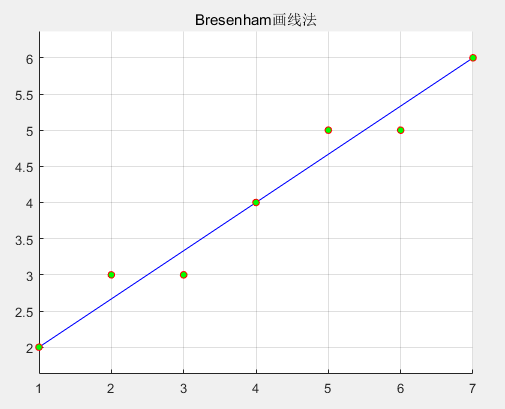


Figure 3 Bresenham画线法

圆绘制的MATLAB代码：

% file:drawCircle.m

% 电子信息工程2017-4 #21 王新钰

% 2020年3月10日17点27分

% lab1:绘制以r=4为半径，(8,8)为圆心的圆，不可直接调用MATLAB中画圆的函数。

% 本程序调用matlab自带函数仅用于对比

close all

clear

%方法1：中点画圆法

figure(1)

midpointCircle(8,8,4)

%方法2：Bresenham画圆法

figure(2)

BresenhamCircle(8,8,4)

%方法3：正负画圆法

figure(3)

pnCircle(xc,yc,r)

function midpointCircle(xc,yc,r)

% 中点法画圆

%figure;

axis equal;

grid on;

rectangle('Position',[xc-r,yc-r,2\*r,2\*r],'Curvature',[1,1],'linewidth',1)

title('MidpointCircle');

x=0;

y=r;

d=1.25-r;

circleEighth(xc,yc,x,y);

while x < y

    if d<0

        d=d+2\*x+3;

    else

        d=d+2\*(x-y)+5;

        y=y-1;

    end

    x=x+1;

    circleEighth(xc,yc,x,y);

end

function BresenhamCircle(xc,yc,r)

% Bresenham法画圆

%figure;

axis equal;

grid on;

rectangle('Position',[xc-r,yc-r,2\*r,2\*r],'Curvature',[1,1],'linewidth',1)

title('BresenhamCircle');

x=0;y=r;d=3-2\*r;

circleEighth(xc,yc,x,y);

while x < y

    if d<0

        d=d+4\*x+6;

    else

        d=d+4\*(x-y)+10;

        y=y-1;

    end

    x=x+1;

    circleEighth(xc,yc,x,y);

end

function pnCircle(xc,yc,r)

%正负画圆法

axis equal;

grid on;

rectangle('Position',[xc-r,yc-r,2\*r,2\*r],'Curvature',[1,1],'linewidth',1)

title('pnCircle');

x=0;y=r;d=0;

while x<=y

    circleEighth(xc,yc,x,y)

    if d <= 0

        d = d + 2 \* x + 1;

        x = x + 1;

    else

        d = d - 2 \* y + 1;

        y = y-1;

    end

end

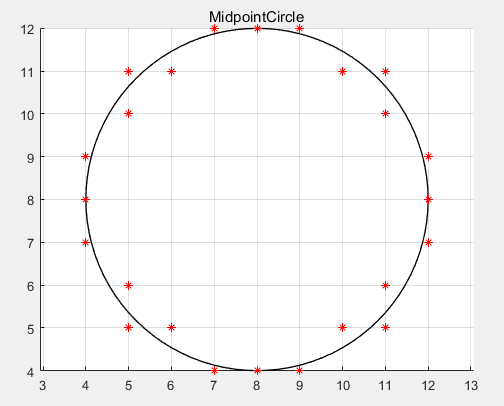


Figure 4 中点画圆法

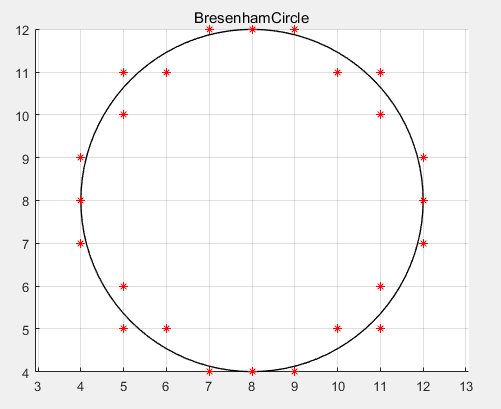


Figure 5 Bresenham法画圆

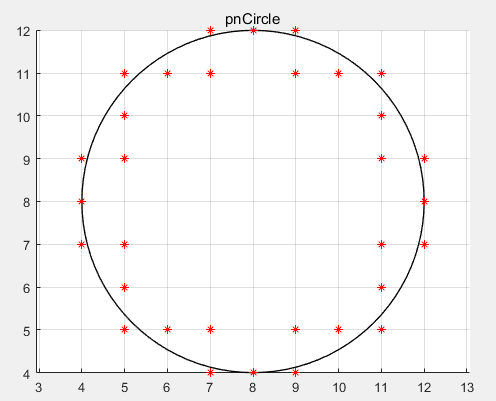


Figure 6 正负画圆法

**五、实验感悟**

通过编程理解了一些基本图形的绘制方法，加深对计算机图形显示学的理解，掌握基本图形的绘制方法。基本理解了如中点画线法，Bresenham画线法，DDA画线法，中点画圆法，Bresenham画圆法，正负画圆法等等。了解了多边形逼近画圆法。采用了MATLAB语言实现对直线及圆用不同的算法进行绘制，加深了对算法的理解。