# 实验一 基本图形设计

## 实验目的与要求

根据教材第二章所学内容，采用基于Html 的javascript 编程实现对金刚石图案和海绵图案设计任务。

基本要求：复现教材第二章中的相关图形参数。

进阶：在教材图形的基础上增加一些创意的设计(比如颜色填充、图形变换等)。

总体要求;实验完成后，把相关程序代码、实验结果截图和实验报告打包上交。

## 实验步骤

1. 创建新html文件，在其中开始写入代码。
2. 创建标签，设定初值。
3. 写入代码，完成图形。
4. 将参考代码中的语句：“context.strokeStyle="#ff0000";”进行了修改：

设置gradient值，将它作为渐变色彩的受体，通过createLinearGradient()函数创建，并通过addColorStop()设置渐变色彩的选择。使context.strokeStyle=gradient，完成图形色彩的替换。

var gradient=context.createLinearGradient(0,0,170,0);

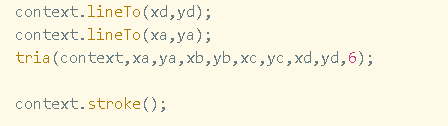
gradient.addColorStop("0","magenta");

gradient.addColorStop("0.5","blue");

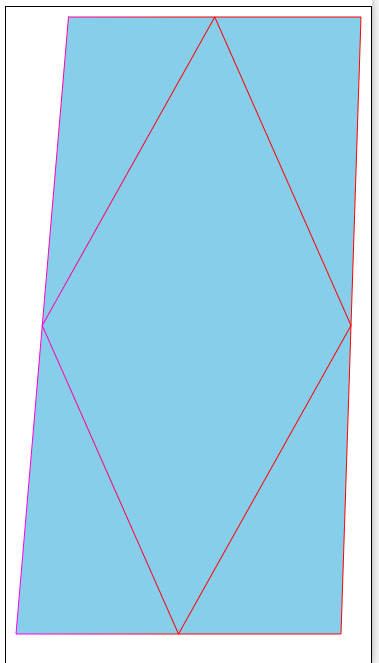
gradient.addColorStop("1.0","red");

context.strokeStyle=gradient;

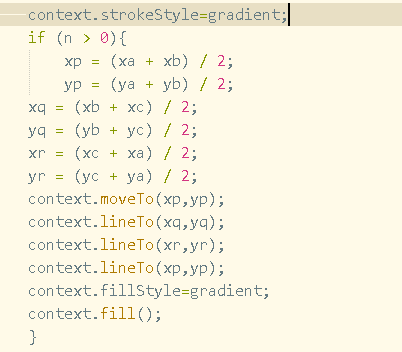
1. 通过改变参考代码中坐标点的位置（海绵图）、或r等值的长度来改变图形的形状、大小。
2. 通过在tria()循环函数中添加fill()语句，可以为三角形填充颜色.
3. 
4. 当在海绵图中添加一个点，形成四边形时，无法进行多次迭代。



结果：（这是基于数学事实的）



1. 当在tria()循环中填充色彩时，是以全部图形进行填充的。



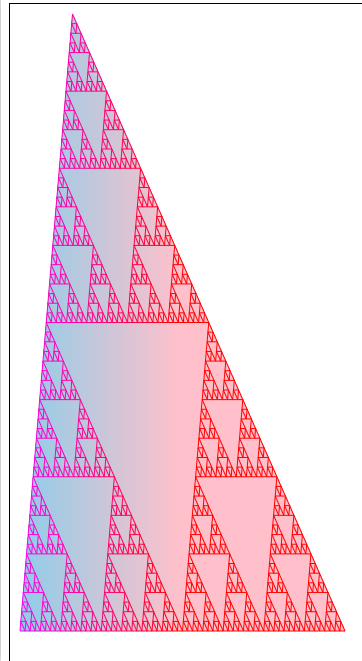
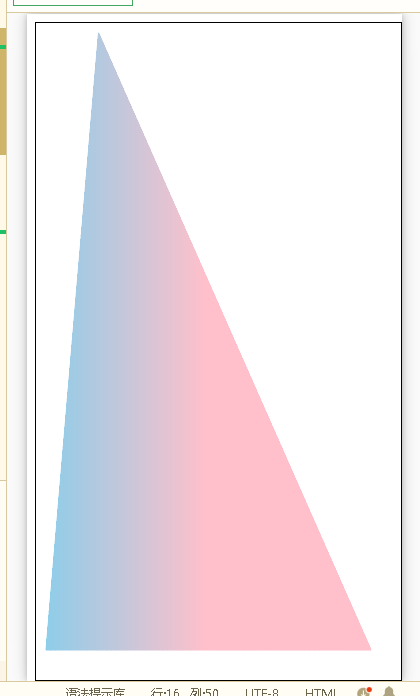
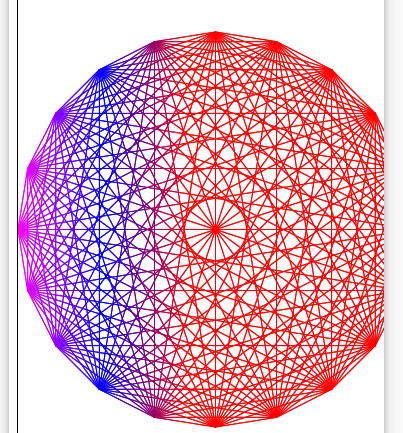
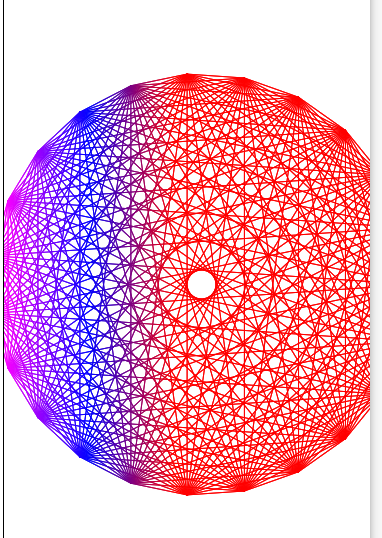
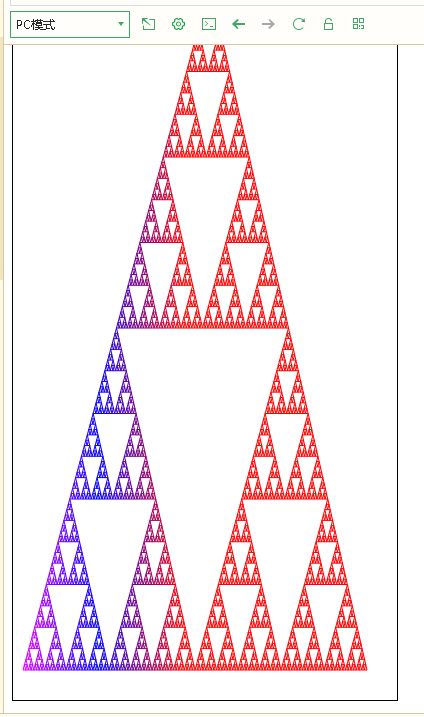
结果：

图1：先stroke再fill；图2：先fill再stroke。

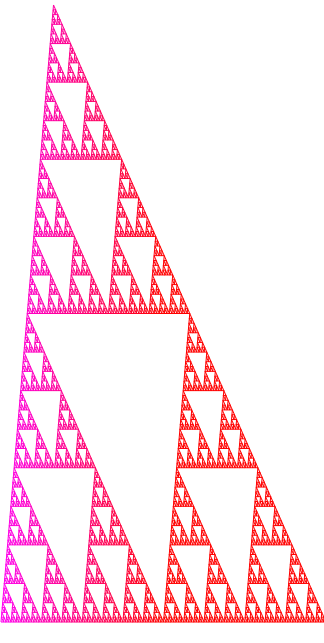
1. （金刚石）当N的值为偶数，图案会穿过圆心。（基于数学原理）



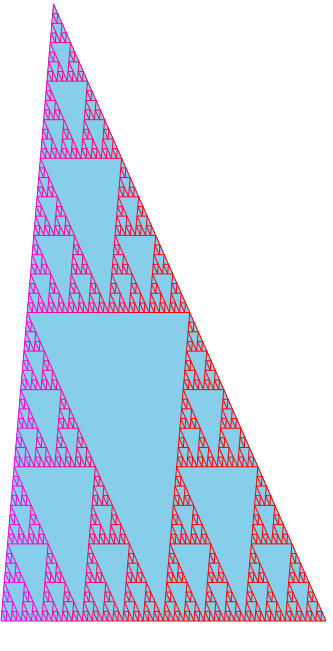
## 实验结果



调整三角形顶点的x坐标后：



填充色彩后：



## 结果分析与结论

1.两张图案的绘制均通过context.moveTo()函数一笔画出；其基于点间连线形成图案。当点的数量为质数与非质数时，连线方式略有不同。通过设置不同的函数，找到目标点并将两点之间连线，我们可以得到目标的图形。

图形的本质来自于两点之间的连线；而设置恰当的点的位置的函数，可以帮助我们画出不同的图形。

2.关于fill()填充结果的解释：

在[Canvas](https://so.csdn.net/so/search?q=Canvas&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/zwt_1/article/details/_blank)中使用fill()进行填充，遵循非零环绕填充规则。

非零环绕填充规则：如果想知道某一区域是否被填充，就从这一区域画一条直线向外，在与其他线的交点处，如果其他线是顺时针方向画的，就+1，逆时针则-1，如果最后总和为0，则不会填充，非零则填充。