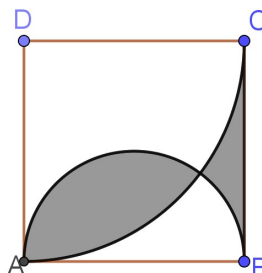



例谈 GeoGebra 涂色十法

浙江萧若茂 313077977

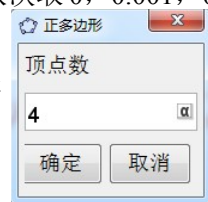
在 GeoGebra 中，经常会遇到涂色问题，GeoGebra 涂色方法灵活，可以根据实际问题选择合适的方法。本文结合实例对各种涂色方法介绍如下：

【例】在正方形 ABCD 中，以 AB 为直径在正方形内部作半圆，以 D 为圆心，以 DA 为半径作圆弧 AC，请将半圆 AB 弧 AC 和线段 BC 所围成的区域进行涂色，如右图所示。



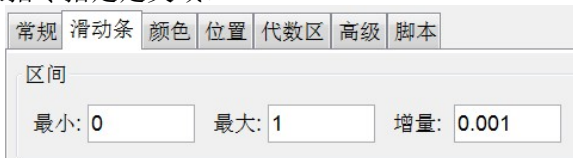
为便于作图，先左键点选正多边形工具，在绘图区依次左键点击 A(0,0)，B(2,0)在弹出的对话框内输入定点数为 4 并确定，生成正方形 ABCD。

法 1（序列线段法）：指令栏输入：序列(线段((t, 2 - sqrt(4 - t^2)), (t, sqrt(1 - (t - 1)^2))), t, 0, 2, 0.01)并回车。此法生成 201 条与边 AB 垂直的线段，形成阴影区域。指令含义是：t 依次取 0, 0.001, 0.002, ……，1.999, 2，对应生成以(t, 2 - sqrt(4 - t^2)), (t, sqrt(1 - (t - 1)^2))为端点的 201 条线段，此法需要求出阴影区域边界的函数表达式，此法根本思想是序列指令。



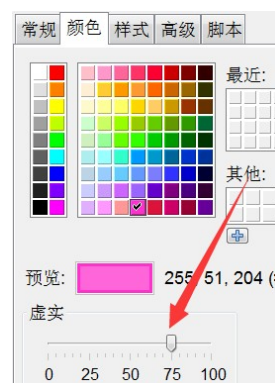
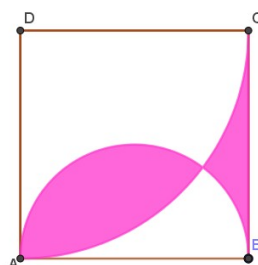
法 2（序列函数法）：指令栏输入：序列(如果(0 ≤ x ≤ 2, (1 - t) (2 - sqrt(4 - x^2)) + t sqrt(1 - (x - 1)^2))), t, 0, 1, 0.05)并回车。此法生成 21 条由上下边界两个函数与 t 共同确定的函数图像，形成阴影区域。指令含义是：t 依次取 0, 0.005, 0.010, ……，0.995, 1，对应生成函数 y=(1 - t) (2 - sqrt(4 - x^2)) + t sqrt(1 - (x - 1)^2)的图像，其中定义域为 0 ≤ x ≤ 2，此使用序列指令生成 21 条函数图像，并用如果指令指定定义域。

法 3（跟踪函数法）：（1）建立滑动条 t，指定其范围为 0-1，并在其属性设置增量为 0.001；（2）然后在指令栏输入：如果(0 ≤ x ≤ 2, (1 - t) (2 - sqrt(4 - x^2)) + t sqrt(1 - (x - 1)^2))并回车，右键选择



画出的函数图像，选择跟踪；（3）右键点击滑动条 t，选择启动动画；屏幕上出现跟踪参数 t 决定的函数图像形成的阴影区域。本方法提高了使用跟踪涂色的方法，本法适当变动可以改为跟踪线段法，读者可以自行尝试。

法 4（轨迹法）：（1）在指令栏内输入：{d, c, 线段(B, C)}并回车，生成以弧 AC,半圆 AB,线段 BC 组成的列表 L_2；其中 d, c 分别是弧 AC,半圆 AB 的名称，GeoGebra 里每一个对象都有唯一一个有别于其他对象的名称，作为其标识符；（3）在指令栏内输入：描点(L_2)并回车，生成列表上的一个自由点 E；（4）在指令栏内输入：F=E 并回车，生成一个与 E 重合的点 F；（5）在指令栏内输入：轨迹(F, E)并回车，生成由 E 决定的点 F 的轨迹；（6）右键选中生成的轨迹，在属性选项卡里选择颜色，把虚实拖到 75%，效果图如右所示。



法 5（多边形法）：（1）在指令栏内输入：L_2={d, c, 线段(B, C)}并回车，生成以弧 AC,半圆 AB,线段 BC 组成的列表 L_2；（2）在指令栏内输入：多边形(序列(描点(L_2, i), i, 0, 1, 0.01))并回车，

生成在列表 L_2 上依次取得 101 个点为顶点的多边形，其中第 1 个点和第 101 个点重合，本方法联合使用序列指令、描点指令、多边形指令生成一个 100 边形近似模拟阴影区域。

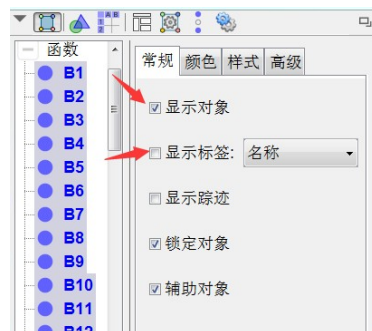
法 6（积分法）：在指令栏内输入：积分介于 $(2 - \sqrt{4 - x^2}, \sqrt{1 - (x - 1)^2}, 0, 2)$ 并回车，生成在区间 $[0, 2]$ 上夹在两个函数 $y_1 = 2 - \sqrt{4 - x^2}$, $y_2 = \sqrt{1 - (x - 1)^2}$ 之间的区域的面积，同时显示这个区域。


法 7（曲面法）：在指令栏内输入：曲面 $(2s + i((1 - t/50)(2 - \sqrt{4 - 4s^2}) + t/50\sqrt{1 - (2s - 1)^2}), s, 0, 1, t, 0, 50)$ 并回车，生成阴影区域。

法 8（不等式法）：在指令栏内输入： $y \geq 2 - \sqrt{4 - x^2} \wedge y \leq \sqrt{1 - (x - 1)^2} \vee y \leq 2 - \sqrt{4 - x^2} \wedge y \geq \sqrt{1 - (x - 1)^2}$ 并回车，生成阴影区域。

法 9（表格法）：（1）在指令栏内输入：L_3=序列(如果 $(0 \leq x \leq 2, (1 - t)(2 - \sqrt{4 - x^2}) + t\sqrt{1 - (x - 1)^2})$, t, 0, 1, 0.05)，生成一个 21 个函数图像组成的列表；（2）打开表格视图，在 A1 至 A21 单元格中依次输入 1 至 21，然后在 B1 单元格中输入：L_3(A1) 并回车，使用类似 EXCEL 的操作方法将 B1 自动填充至在 B21；（3）右键选中 B 列，选择属性，在常规选项卡里勾选【显示对象】，去除【显示标签】。绘图区出现 B1 至 B21 所定义的函数图像。

本方法可以提取列表元素，对列表元素单独设置颜色，是一种重要的着色方法。



法 9（JS 扫描法）：（1）左键选择按钮工具  在绘图区空白处左键单击，在弹出的对话框里设置标题为：JS 扫描，点击确定，生成按钮【JS 扫描】；（2）右键点击按钮【JS 扫描】选择属性在弹出的对话框里，选择脚本，在单击时输入以下代码：

```
var ggb=ggbApplet;
var num=100;
for( var i=0;i <=2*num;i++){
ggb.evalCommand("xd=Segment((" + i/num + ", 2 - sqrt(4 - (" + i/num + ")^2)), (" + i/num + ", sqrt(1 - (" + i/num + ") - 1)^2)))");
ggb.evalCommand("ShowLabel(xd,false)");
ggb.evalCommand("SetVisibleInView(xd,1,true)");
ggb.evalCommand("SetColor(xd,cos((" + i/num + ")*e^(" + i/num + ")),ln((" + i/num + ")))");
ggb.evalCommand("SetTrace(xd,true)");
}
ggb.evalCommand("SetVisibleInView(xd,1,false)");
```

并在对话框下方选择 JavaScript, 点击确定。

左键单击按钮【JS 扫描】，即可生成右图彩色区域。此法利用循环语句生成 101 条线段，每一条线段的色彩由参数控制，此法也是着色的一种通法。

