tkz-euclide

tkz-euclide 学习笔记

作者：Alain Matthes
组织：钢铁理工男
时间：2021 年 7 月 8 日
版本：1.00

不懂几何者，不得入内!

目录

第一章 概述

1.1 什么是 tkz-euclide

tkz-euclide 是一个用于在笛卡尔坐标系中绘制平面图形 (点、线、三角形、圆等基本二维对象) 的尽可能方便使用的宏包，它主要用于绘制经典欧氏几何图形。tkz-euclide 是基于 PGF 设计的，可以看作是一个 TikZ 的前端，并且是符合 LaTeX 语法的绘图宏包。其目标是提供一个高层的用户接口，以通过相对简单语法进行绘图。它使用 tkz-base 宏包提供的正交笛卡尔坐标系和点的定义及点的操作宏命令为基础进行绘图。该宏包的基本思路是尽可能符合欧氏几何中手工绘图的绘图方式。

另，由于 tkz-euclide 开发者 Alain Matthes 的母语是法语（导致部分命令看上去像英文又有点不一样。例如 euclide 就是法语的欧几里得）译者又是根据 tkz-euclide 英语手册翻译的，该说明文档中，可能会存在描述错误。笔记中会根据自己的理解做适当调整

1.2 为什么要开发 tkz-euclide?

Alain Matthes 开发该宏包的最初想法是为自己和其它数学老师设计一个 LaTeX 绘图工具，以实现欧氏几何图形的快速绘制，而不需要费力学习和掌握一门新的绘图语言。当然，tkz-euclide 适用于所有使用 LaTeX 的教师，以实现在 LaTeX 中轻松、正确地欧氏几何绘图。

显然，最简单的绘图方法是按手工绘图的方式和思维进行绘图。为描述一个几何图形，须定义图形对象以及对这些对象所执行的操作。因此，如果与数学家或学习数学的学生使用的数学语言语法相近，这些语法则更容易理解和掌握。当然，宏包的语法也必须符合 LaTeX 用户的使用习惯。

基于此，该宏包定义了点、线段、直线、三角形、多边形和圆六种图形对象。并且设计了定义、创建、绘制、标记和标注五个对图形对象的基本操作

虽然这会使语法比较冗长，但却更容易理解和使用。因此，用户能够轻松使用该宏包提供的命令进行绘图。

1.3 一些建议

- 导言区
  如果需要使用 xcolor 宏包，则必须在 tkz-euclide 之前载入该宏包，也就是在 TikZ 宏包之前载入。否则可能会与 TikZ 宏包冲突，但 TikZ 宏包提供的 babel 库能解决这些冲突。
- tkz-euclide 绘图中的基本要素是点
- 通常的绘图流程
  1. 定义已知点：多数情况下，并不需要使用\tkzInit 和\tkzClip 命令
  2. 利用已知点通过计算得到其它点
  3. 图形绘制
  4. 绘制标记
  5. 布置标注
- 约定和符号
  1. 该宏包选用法国的几何符号和习惯描述几何图形 (谁让作者是法国人呢, 不爽你来呀)
  2. tkz-euclide 宏包定义和表示的对象有平面中的点、线和圆，它们是欧氏几何的主要元素，可以根据这些基本元素构成欧氏几何图形。根据欧几里得的定义，仅使用这些基本图形就可以构成各种复杂图形。因此，一个点并没有具体的尺寸，它在现实中不存在。同样，一条线也没有宽度，也不存在。在 tkz-euclide 宏包中需要考虑的是代表理想数学概念的对象

3. tkz-euclide 遵循古希腊尺规作图的方式进行绘图
4. 点可以用圆盘或十字线表示 (两条直线、一条直线和圆或两个圆的交点)
5. 一个点，可以使用类似 A、B 或 C 这样的大写字母进行标注 (当然，也会有例外 A_1, M_A)。一个点在代码中的引用名称和标注名称可能不一样，所以可以定义一个点 A，但是将其标注为 P(没必要这样做)。另外一种例外情况是无需标注的内部点，这些点在代码中常常用小写字母表示 (a, b_1)
6. 线段使用方括号中的两个点表示，如：`[AB]`
7. 在欧氏几何中，直线用两个点表示，因此，点 A 和点 B 定义的直线表示为 (AB)。也可以使用希腊字母表示直线，并将其命名为 δ, Δ。也可以使用小写字母表示直线，如 d, d'
8. 射线可表示为`[AB)`
9. 直线间的关系。如对于直线 (AB) 和直线 (CD)，垂直表示为 $(AB) \perp (CD)$，平行表示为 $(AB) \,//\, (CD)$
10. 三角形 ABC 的边长表示为 AB、AC 和 BC。长度值一般用小写字母表示如：$AB = c$、$AC = b$ 和 $BC = a$。字母 a 也常常用于表示一个角度，r 常常用于表示半径，d 表示直径，l 表示长度，d 也可以表示距离
11. 多边形用其顶点表示，如三角形表示为 ABC，四边形表示为 $EFGH$。
12. 角度的单位是度 (例如：60°)，对于等边三角形 ABC，可以表示为 $\widehat{ABC} = \widehat{B} = 60°$。
13. 圆弧用起止点表示，如，若 A 和 B 是同一个圆上的两个点，则可以用 $\overset{\frown}{AB}$ 表示圆弧。
14. 如果没有岐义，一个圆可以表示为 $\mathcal{C}$，或用 $\mathcal{C}(O\,;\,A)$ 表示圆心在 O 点并通过点 A 的圆或用 $\mathcal{C}(O\,;\,1)$ 表示圆心在点 O 半径为 1 cm 的圆。
15. 三角形中的特殊线有：内角角平分线、外角角平分线等。
16. (x_1,y_1) 表示点 A_1 的坐标分量，(x_A,y_A) 表示点 A 的坐标分量。

1.4 使用 tkz-euclide 宏包经典示例

在此，以绘制一个等边三角形为例，展示 tkz-euclide 宏包的正确使用方式。当然，绘制该图形可以有多种方式，本例中将遵循欧氏几何尺规绘图步骤。

- 首先需要引入文档类，对于单个图形而言，比较方便的一种方式是使 standalone 文档类

```
\documentclass{standalone}
```

- 然后载入 tkz-euclide 宏包：

```
\usepackage{tkz-euclide}
```

注意，由于 tkz-euclide 宏包是基于 TikZ 宏包开发的，会同时载入该宏包，因此，无需再次载入 TikZ 宏包

- 开始文档，并使用 tikzpicture 环境绘制欧氏几何图形：

```
\begin{document}
\begin{tikzpicture}
```

- 定义两个已知点：

```
\tkzDefPoint(0,0){A}
\tkzDefPoint(5,2){B}
```

- 使用这两个点定义两个圆，并使用这两个圆定义交点：(A, B) 表示以 A 点为圆心通过 B 点，(B, A) 表示以 B 点为圆心通过 A 点，两个圆共用这 A 点和 B 点。

```
\tkzInterCC(A,B)(B,A)
```

- 得到两个圆的交点，并命名为 C 和 D

```
\tkzGetPoints{C}{D}
```

- 至此，便完成了所有点的定义，接下来进行绘图。

```
\tkzDrawCircles[gray,dashed](A,B B,A)
\tkzDrawPolygon(A,B,C)% 绘制多边形
```

- 绘制点 A、B、C 和 D：

```
\tkzDrawPoints(A,...,D)
```

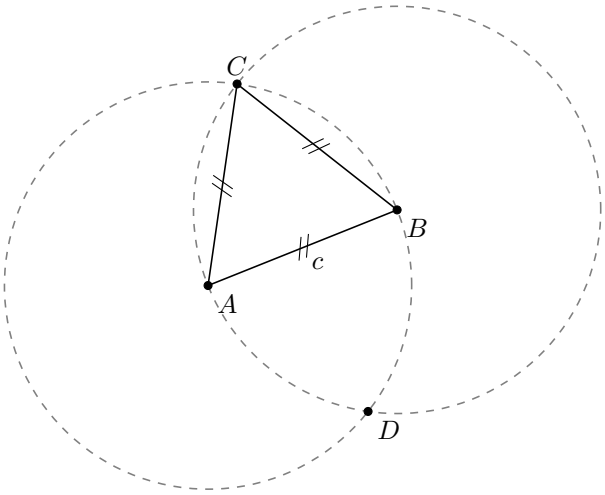
- 绘制标注，在绘制标注时，可以为其指定位置参数。

```
\tkzLabelSegments[swap](A,B){$c$}
\tkzLabelPoints(A,B,D)
\tkzLabelPoints[above](C)
```

- 最后，结束各个环境

```
\end{tikzpicture}
\end{document}
```

- 完整的代码




```
\begin{tikzpicture}[scale=0.5]
% 定义点
\tkzDefPoint(0,0){A}
\tkzDefPoint(5,2){B}
% 计算得到的点
\tkzInterCC(A,B)(B,A)
\tkzGetPoints{C}{D}
% 绘图
\tkzDrawCircles[gray,dashed](A,B B,A)
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,...,D)
% 标记
\tkzMarkSegments[mark=s||](A,B B,C C,A)
% 标注
\tkzLabelSegments[swap](A,B){$c$}
\tkzLabelPoints(A,B,D)
\tkzLabelPoints[above](C)
\end{tikzpicture}
```

第二章 点的定义 — 坐标点

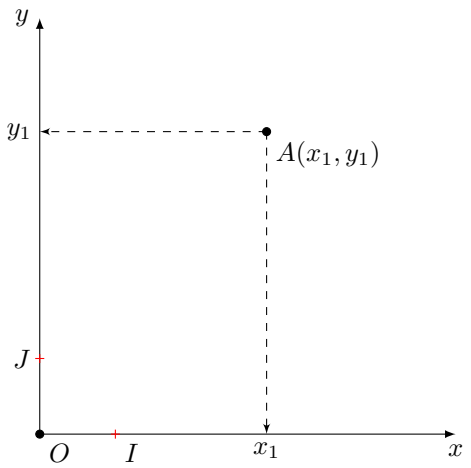
可以通过如下方式定义一个点:

- 笛卡尔坐标
- 极坐标
- 已命名的点
- 相关的点

点可以用一对十进制数进行定义。因此，可以用 (x, y) 或 $(a : d)$ 表示平面中的一个点。其中 (x, y) 是笛卡尔坐标的写法：x 是横坐标，逗号,是分隔符，y 是纵坐标；$(a : d)$ 是极坐标的写法：a 是角度，冒号:是分隔符，d 是距离；其中度量单位是度和厘米。虽然仅用坐标就可以定义一个点，但强烈建议为点进行命名。

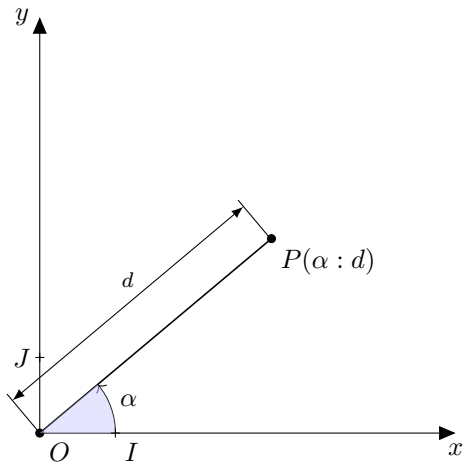
请参阅以下两段代码及其绘制结果，以体会这些命令的作用。

代码 1：笛卡尔坐标




```
\begin{tikzpicture}[scale=1]
\tkzInit[xmax=5,ymax=5]
\tkzDefPoints{0/0/O,1/0/I,0/1/J,3/4/A}
\tkzDrawXY[noticks,>=latex]
\tkzDrawPoints(O,A)
\tkzLabelPoint(A){$A(x_1,y_1)$}
\tkzShowPointCoord[xlabel=$x_1$,ylabel=$y_1$](A)
\tkzLabelPoints(O,I)
\tkzLabelPoints[left](J)
\tkzDrawPoints[shape=cross,red](I,J)
\end{tikzpicture}
```

代码 2：极坐标




```
\begin{tikzpicture}[scale=1]
\tkzInit[xmax=5,ymax=5]
\tkzDefPoints{0/0/O,1/0/I,0/1/J}
\tkzDefPoint(40:4){P}
\tkzDrawXY[noticks,>=triangle 45]
\tkzDrawSegment[dim={$d$,16pt,above=6pt}](O,P)
\tkzDrawPoints(O,P)
\tkzMarkAngle[mark=none,->](I,O,P)
\tkzFillAngle[fill=blue!20,opacity=.5](I,O,P)
\tkzLabelAngle[pos=1.25](I,O,P){$\alpha$}
\tkzLabelPoint(P){$P (\alpha : d )$}
\tkzDrawPoints[shape=cross](I,J)
\tkzLabelPoints(O,I)
\tkzLabelPoints[left](J)
\end{tikzpicture}
```

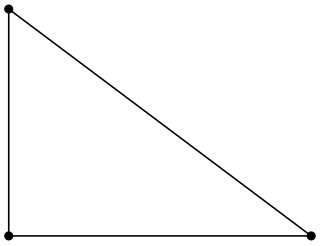
2.1 定义坐标点

可以用\tkzDefPoint 命令，通过指定坐标定义一个点。该命令基于 TikZ 的\coordinate 命令实现，因此，可在该命令中使用类似 shift 的 TikZ 的选项，该命令使用 xfp 宏包实现计算。在定义点时，既可以使用笛卡尔坐标，也可以使用极坐标。

\tkzDefPoint[选项](x,y){名称} or $(\alpha:d)${名称}

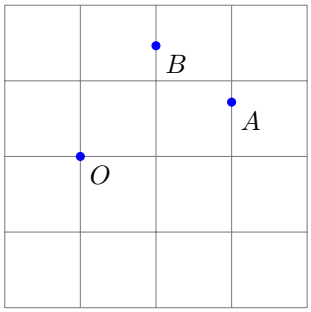
参数	默认值	含义
(x,y)	无	x 和 y 分别是 2 维坐标，默认单位是 cm
$(\alpha:b)$	无	α 是角度 (度), d 是距离 (cm)
{名称}	无	点的名称，如：$A, T_a, P1, \ldots$
必选参数是十进制表示的 2 维坐标值，笛卡尔坐标表示两个长度，极坐标表示角度和距离		
选项	**默认值**	**含义**
label	无	按预设的距离添加标注
shift	无	为 (x,y) 或 $(\alpha:d)$ 添加坐标偏移

2.1.1 笛卡尔坐标示例



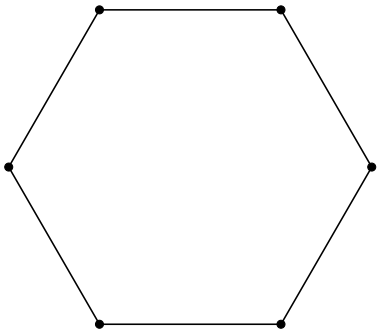
```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=5]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDefPoint(0,3){C}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
\end{tikzpicture}
```

2.1.2 使用 xfp 宏包实现计算




```
\begin{tikzpicture}[scale=1]
\tkzInit[xmax=4,ymax=4]
\tkzGrid
\tkzDefPoint(-1+2,sqrt(4)){O}
\tkzDefPoint({3*ln(exp(1))},{exp(1)}){A}
\tkzDefPoint({4*sin(pi/6)},{4*cos(pi/6)}){B}
\tkzDrawPoints[color=blue](O,B,A)
\tkzLabelPoints(O,A,B)
\end{tikzpicture}
```

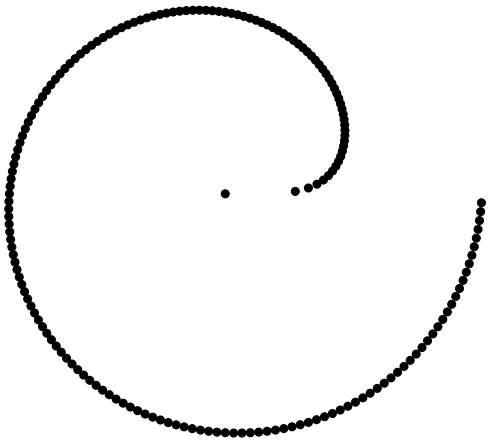
2.1.3 极坐标示例



```
\begin{tikzpicture}[scale=0.8]
\foreach \an [count=\i] in {0,60,...,300}{
\tkzDefPoint(\an:3){A_\i}
}
\tkzDrawPolygon(A_1,A_...,A_6)
\tkzDrawPoints(A_1,A_...,A_6)
\end{tikzpicture}
```

2.1.4 坐标计算

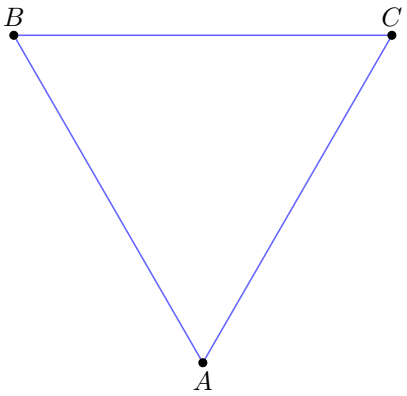
在计算坐标时，需遵循 xfp 宏包语法。另外，如使用 pgfmath 库计算，则必须在使用\tkzDefPoint 命令前完成计算。



```
\begin{tikzpicture}[scale=.6]
\foreach \an [count=\i] in {0,2,...,358}{
\tkzDefPoint(\an:sqrt(sqrt(\an mm))){A_\i}
}
\tkzDrawPoints(A_1,A_...,A_180)
\end{tikzpicture}
```

2.1.5 相对位置点

可以使用相对位置定义一个点，此时，需使用 TikZ 的 scope 环境。
下面的代码给出了一种定义等边三角形的方法：




```
\begin{tikzpicture}[scale=1]
\tkzSetUpLine[color=blue!60]
\begin{scope}[rotate=30]
\tkzDefPoint(2,3){A}
\begin{scope}[shift=(A)]
\tkzDefPoint(90:5){B}
\tkzDefPoint(30:5){C}
\end{scope}
\end{scope}
\tkzDrawPolygon(A,B,C)
\tkzLabelPoints[above](B,C)
\tkzLabelPoints[below](A)
\tkzDrawPoints(A,B,C)
\end{tikzpicture}
```

2.2 定义相对坐标点

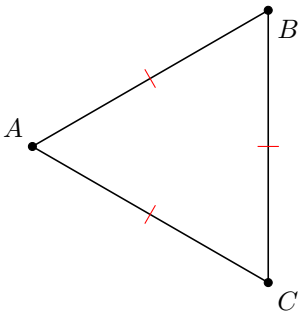
\tkzDefShitPoint[参考点](x,y){名称} 或 $(\alpha:d)${名称}

参数	默认值	含义
(x,y)	无	x 和 y 是 2 维坐标，默认单位是 cm
$\alpha:d$	无	α 是角度 (度)，d 是距离
选项	默认值	含义
参考点	无	例如：\tkzDefShiftPoint[A](0:4){B}

\tkzDefShiftPoint 命令和*\tkzDefShiftPointCoord* 命令非常像，感觉区别是 *\tkzDefShiftPoint* 的参考点是一个点的名字，*\tkzDefShiftPointCoord* 的参考点是一组坐标数字

2.2.1 构造等边三角形

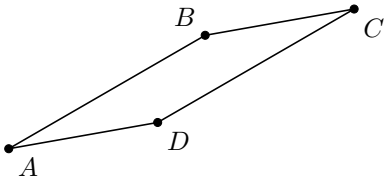
下面的代码给出了一种极为简捷的等边三角形构造方法。




```
\begin{tikzpicture}[scale=1.2]
\tkzDefPoint(2,3){A}
\tkzDefShiftPoint[A](30:3){B}
\tkzDefShiftPoint[A](-30:3){C}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(B,C)
\tkzLabelPoints[above left](A)
\tkzMarkSegments[mark=|,color=red](A,B A,C B,C)
\end{tikzpicture}
```

2.2.2 构造平行四边形

简单的定义平行四边形的方式为：



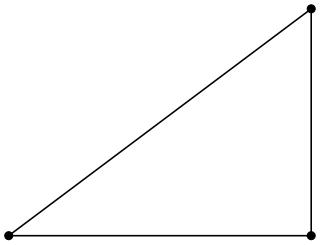

```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(30:3){B}
\tkzDefShiftPointCoord[B](10:2){C}
\tkzDefShiftPointCoord[A](10:2){D}
\tkzDrawPolygon(A,...,D)
\tkzDrawPoints(A,...,D)
\tkzLabelPoints(A,C,D)
\tkzLabelPoints[above left](B)
\end{tikzpicture}
```

2.3 定义点集

\tkzDefPoints[命令选项]{ $x_1/y_1/n_1, x_2/y_2/n_2, \cdots$ }

参数	默认值	样例
$x_i/y_i/n_i$	无	x_i 和 y_i 是 n_i 点的 2 维坐标
选项	**默认值**	**含义**
shift	无	为所有点添加 (x, y) 或 $(\alpha : d)$ 坐标偏移

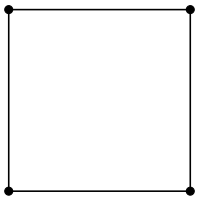
2.3.1 构造三角形



```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/A,4/0/B,4/3/C}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
\end{tikzpicture}
```

2.3.2 构造正方形

注意该代码中绘制多边形的语法。



```
\begin{tikzpicture}[scale=1.2]
\tkzDefPoints{0/0/A,2/0/B,2/2/C,0/2/D}
\tkzDrawPolygon(A,...,D)
\tkzDrawPoints(A,B,C,D)
\end{tikzpicture}
```

2.4 绘制点

2.4.1 绘制单点

\tkzDrawPoint[选项](点名称)

选项	默认值	含义
shape	circle	点的样式，可以使用 cross 或 cross out。我们还可以自定义点的样式
size	6	6 ×\pgflinewidth
color	black	默认颜色，可以修改。点圆盘用填充色绘制，可以通过 fill,draw 选项实现更多效果

注意：*tikzpicture* 环境的缩放参数 *scale* 不会影响点的大小，可以用点绘制命令中的 *size* 参数改变点的大小

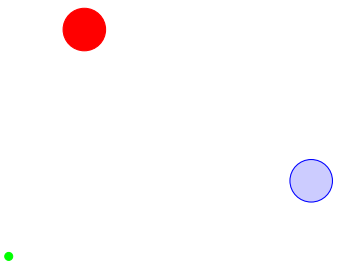
2.4.2 \tkzDrawPoints命令：绘制多点

绘制多点和绘制单点的命令类似。但要注意不要忘记命令末尾的 s 字母 (西语特色)。有一点需要关注的是，绘制多个点时，所有点使用相同参数

手册中很多命令都分单数、复数形式。下文中，在选项、参数非常近似的情况下会同时描述

2.4.3 点绘制示例

示例一：绘制单点

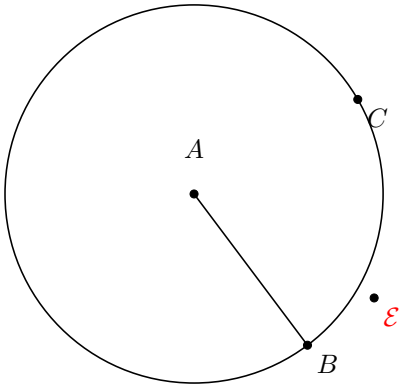


```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{1/3/A,4/1/B,0/0/O}
\tkzDrawPoint[color=red,size=16pt](A)
\tkzDrawPoint[fill=blue!20,draw=blue,size=16pt](B)
\tkzDrawPoint[color=green](O)
\end{tikzpicture}
```

示例二：绘制多点

```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(1,3){A}
\tkzDefPoint(4,1){B}
\tkzDefPoint(0,0){C}
\tkzDrawPoints[size=6,color=red,fill=red!50](A,B,C)
\end{tikzpicture}
```

示例三：




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(2,3){A}
\tkzDefPoint(5,-1){B}
\tkzDefShiftPoint[A](-30:5.5){E}
\begin{scope}[shift=(A)]
\tkzDefPoint(30:5){C}
\end{scope}
\tkzCalcLength[cm](A,B)
\tkzGetLength{rAB}
\tkzDrawCircle[R](A,\rAB cm)
\tkzDrawSegment(A,B)
\tkzDrawPoints(A,B,C,E)
\tkzLabelPoints(B,C)
\tkzLabelPoints[above=10pt](A)
\tkzLabelPoint[red](E){$\mathcal{E}$}
\end{tikzpicture}
```

2.5 标注点

2.5.1 标注单点

\tkzLabelPoint[选项](点名称){标注文本}

参数	默认值	含义
点名称	无	只能有一个点的名称
标注文本	点名称	LaTeX 文本或行间数学环境，缺省使用点名称标注
选项	**默认值**	**含义**
标注位置	below right	可单独或组合使用 above,below,left,right
标注距离		例如：above=10pt
color	black	默认颜色，可以修改。例如 red,color=teal
font		字体大小。例如：font=\scriptsize

2.5.2 标注点集

标注点集和标注单点的命令类似。但要注意不要忘记命令末尾的s字母 (西语特色)。有一点需要关注的是，标注多个点时，所有点使用相同参数

\tkzLabelPoints[选项](点名称列表)

选项	默认值	含义
标注位置	below right	可单独或组合使用 above,below,left,right
标注距离		例如：above=10pt
color	black	默认颜色，可以修改。例如 red,color=teal
font		字体大小。例如：font=\scriptsize

第三章 点的定义 — 特殊位置点

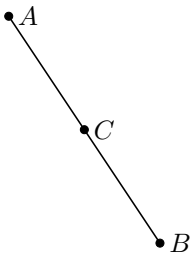
3.1 定义线段中点

\tkzDefMidPoint($pt1, pt2$)

定义的点存储于\tkzPointResult 命令中，也可以通过\tkzGetPoint 命令得到该点，并为其命名

参数	默认值	含义
$pt1, pt2$	无	$pt1$ 和 $pt2$ 是线段的两个端点

3.1.1 \tkzDefMidPoint 命令示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(2,3){A}
\tkzDefPoint(4,0){B}
\tkzDefMidPoint(A,B) \tkzGetPoint{C}
\tkzDrawSegment(A,B)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints[right](A,B,C)
\end{tikzpicture}
```

3.2 定义重心点

设共有 $pt_1, pt_2, \ldots, pt_n$ 共 $n(n \geq 2)$ 个点，则它们定义了 n 个向量 $\overrightarrow{v_1}, \overrightarrow{v_2}, \ldots, \overrightarrow{v_n}$。令 α_1,α_2,$\ldots \alpha_n$ 是 n 常数, 因此可按下式得到一个新向量：

$$\frac{\alpha_1\overrightarrow{v_1} + \alpha_2\overrightarrow{v_2} + \cdots + \alpha_n\overrightarrow{v_n}}{\alpha_1 + \alpha_2 + \cdots + \alpha_n}$$

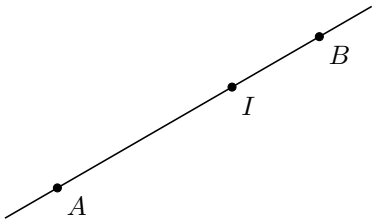
则，该向量可用于定义一个点。这个点就是 $pt_1, pt_2, \ldots, pt_n$ 的重心

\tkzDefBarycentricPoint($pt1 = a_1, pt2 = a_2, \ldots$)

参数	默认值	含义
$pt1 = a_1, pt2 = a_2, \ldots$	无	a 是每个点的权重

3.2.1 用\tkzDefBarycentricPoint 命令计算 2 个点的重心

下面的代码中，通过系数 “1” 和 “2” 得到了点 A 和点 B 的重心：$\overrightarrow{AI} = \frac{2}{3}\overrightarrow{AB}$

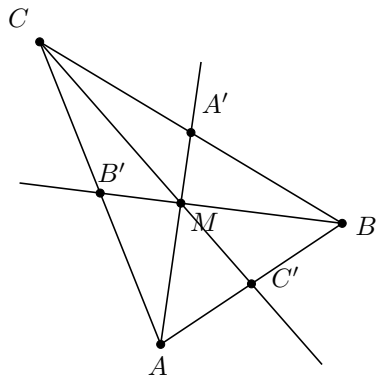



```
\begin{tikzpicture}
\tkzDefPoint(2,3){A}
\tkzDefShiftPointCoord[2,3](30:4){B}
\tkzDefBarycentricPoint(A=1,B=2)
\tkzGetPoint{I}
\tkzDrawPoints(A,B,I)
\tkzDrawLine(A,B)
\tkzLabelPoints(A,B,I)
\end{tikzpicture}
```

3.2.2 用\tkzDefBarycentricPoint 命令计算 3 个点的重心

下面的代码中，M 是三角形的重心。

为简化操作，该宏包还提供了另外一个用于直接计算三角形重心的\tkzCentroid 命令。

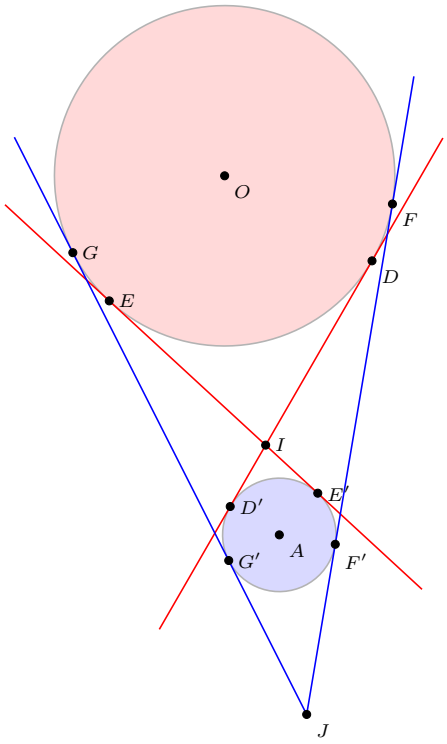



```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(2,1){A}
\tkzDefPoint(5,3){B}
\tkzDefPoint(0,6){C}
\tkzDefBarycentricPoint(A=1,B=1,C=1)
\tkzGetPoint{M}
\tkzDefMidPoint(A,B)  \tkzGetPoint{C'}
\tkzDefMidPoint(A,C)  \tkzGetPoint{B'}
\tkzDefMidPoint(C,B)  \tkzGetPoint{A'}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A',B',C')
\tkzDrawPoints(A,B,C,M)
\tkzDrawLines[add=0 and 1](A,M B,M C,M)
\tkzLabelPoint(M){$M$}
\tkzAutoLabelPoints[center=M](A,B,C)
\tkzAutoLabelPoints[center=M,above right](A',B',C')
\end{tikzpicture}
```

3.3 圆的内外位拟点

\tkzDefIntSimilitudeCenter$(O_1,r_1)(O_2,r_2)$---内位拟点
\tkzDefExtSimilitudeCenter$(O_1,r_1)(O_2,r_2)$---外位拟点

参数	默认值	含义
O	无	圆心
r	无	半径




```
\begin{tikzpicture}[scale=.75,rotate=-30]
\tkzDefPoint(0,0){O}
\tkzDefPoint(4,-5){A}
% 内部相似中心
\tkzDefIntSimilitudeCenter(O,3)(A,1)
\tkzGetPoint{I}
% 外部相似中心
\tkzExtSimilitudeCenter(O,3)(A,1)
\tkzGetPoint{J}
\tkzDefTangent[from with R = I](O,3 cm)
\tkzGetPoints{D}{E}
\tkzDefTangent[from with R = I](A,1 cm)
\tkzGetPoints{D'}{E'}
\tkzDefTangent[from with R = J](O,3 cm)
\tkzGetPoints{F}{G}
\tkzDefTangent[from with R = J](A,1 cm)
\tkzGetPoints{F'}{G'}
\tkzDrawCircle[R,fill=red!50,opacity=.3](O,3 cm)
\tkzDrawCircle[R,fill=blue!50,opacity=.3](A,1 cm)
\tkzDrawSegments[add = .5 and .5,color=red](D,D' E,E')
\tkzDrawSegments[add= 0 and 0.25,color=blue](J,F J,G)
\tkzDrawPoints(O,A,I,J,D,E,F,G,D',E',F',G')
\tkzLabelPoints[font=\scriptsize](O,A,J,D,F,F')
\tkzLabelPoints[font=\scriptsize,right](I,E,G,D',E',G')
\end{tikzpicture}
```

第四章 点的定义 — 三角形的点

定义与三角形相关的几个特殊点

\tkzDefTriangleCenter[选项](A, B, C)

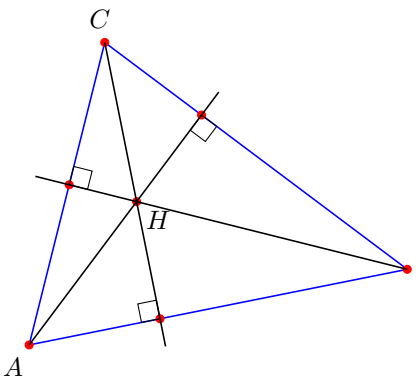
♡ 注意：该命令的参数必须是一个三角形的 3 个顶点列表，结合\tkzGetPoint 命令保存定义的点，并为其命名。当然，仅临时使用，则可使用\tkzPointResult 命令使用该点

参数	默认值	含义
A, B, C	无	逗号分隔的三角形 3 个顶点列表

选项	默认值	含义
ortho	circum	垂心，三条高的交点
centroid	circum	重心，三条中线的交点
circum	circum	外心，外接圆圆心
in	circum	内心，内切圆圆心
ex	circum	旁心，与三角形的一边和其他两边的延长线相切的圆的圆心
euler	circum	欧拉点，欧拉圆/费尔巴哈圆/九点圆圆心
symmedian	circum	陪位重心，Lemoine 点或中间点或 Grebe 点
spieker	circum	Spieker 点，中点三角形内切圆圆心
nagel	circum	Nagel 点 (界心)，三个旁切圆切点与对应顶点连线的交点
mittenpunkt	circum	三个旁切圆圆心与对应边中点连线的交点
feuerbach	circum	Feuerbach 点，内切圆与九点圆的公切点

4.1 ortho 或 orthic 选项：垂心

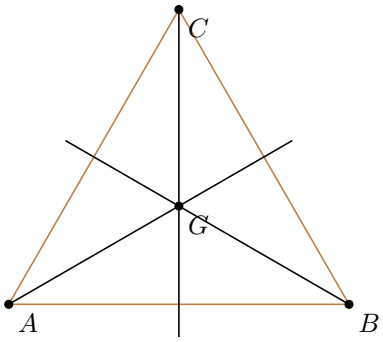
三角形三条高的交点称为三角形的垂心。




```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,5/1/B,1/4/C}
%\tkzClipPolygon(A,B,C)
\tkzDefTriangleCenter[ortho](B,C,A)
\tkzGetPoint{H}
\tkzDefSpcTriangle[orthic,name=H](A,B,C){a,b,c}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawPoints[red](A,B,C,H,Ha,Hb,Hc)
\tkzDrawLines[add=0 and .1](A,Ha B,Hb C,Hc)
\tkzLabelPoint(H){$H$}
\tkzAutoLabelPoints[center=H](A,B,C)
\tkzMarkRightAngles(A,Ha,B B,Hb,C C,Hc,A)
\end{tikzpicture}
```

4.2 centroid 选项：重心

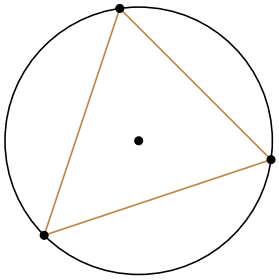
三角形三条中线的交点称为重心。




```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{-1/1/A,5/1/B}
\tkzDefEquilateral(A,B)
\tkzGetPoint{C}
\tkzDefTriangleCenter[centroid](A,B,C)
\tkzGetPoint{G}
\tkzDrawPolygon[color=brown](A,B,C)
\tkzDrawPoints(A,B,C,G)
\tkzDrawLines[add = 0 and 2/3](A,G B,G C,G)
\tkzLabelPoints(A,B,C,G)
\end{tikzpicture}
```

4.3 circum 选项：外心

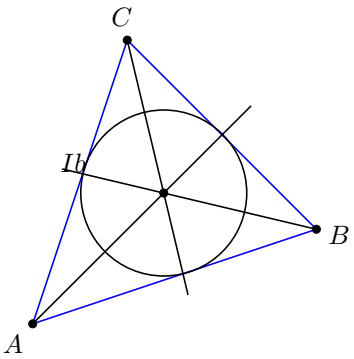
三角形外接圆圆心称为外心，也是三个边垂直平分线的交点。



```
\begin{tikzpicture}
\tkzDefPoints{0/1/A,3/2/B,1/4/C}
\tkzDefTriangleCenter[circum](A,B,C)
\tkzGetPoint{G}
\tkzDrawPolygon[color=brown](A,B,C)
\tkzDrawCircle(G,A)
\tkzDrawPoints(A,B,C,G)
\end{tikzpicture}
```

4.4 in 选项：内心

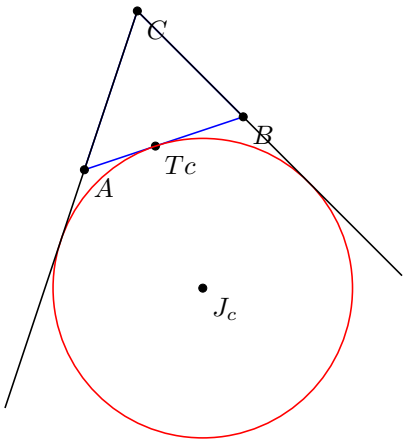
几何学中，三角形的内切圆是三角形内最大的圆，内切圆的圆心称为三角形的内心。三角形的内心也是三角形三个内角角平分线的交点。(https://en.wikipedia.org/wiki/Incircle_and_excircles_of_a_triangle)




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints{0/1/A,3/2/B,1/4/C}
\tkzDefTriangleCenter[in](A,B,C)
\tkzGetPoint{I}
\tkzDefPointBy[projection=onto A-C](I)
\tkzGetPoint{Ib}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawPoints(A,B,C,I)
\tkzDrawLines[add = 0 and 2/3](A,I B,I C,I)
\tkzDrawCircle(I,Ib)
\tkzAutoLabelPoints[center=I](A,B,C,Ib)
\end{tikzpicture}
```

4.5 ex 选项：旁心

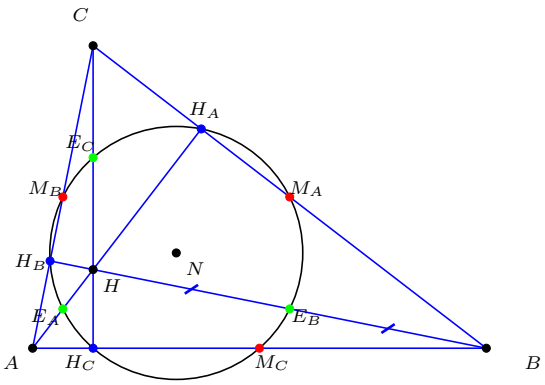
旁切圆圆心是一个顶点 (例如顶点 A) 的内角角平分线与另外两个外角角平分线的交点。该旁切圆圆心是相对于顶点 A 的一个旁心，或叫作 A 的旁心。因为三角形内角角平分线与对应的外角角平分线垂直，所以，内心与 3 个旁心构成了一个正交系统。旁切圆是位于三角形外部与某条边及另外两条边的延长线相切的圆，一个三角形有 3 个旁切圆。(https://en.wikipedia.org/wiki/Incircle_and_excircles_of_a_triangle)




```
\begin{tikzpicture}[scale=0.70]
\tkzDefPoints{0/1/A,3/2/B,1/4/C}
\tkzDefTriangleCenter[ex](B,C,A)
\tkzGetPoint{J_c}
\tkzDefPointBy[projection=onto A-B](J_c)
\tkzGetPoint{Tc}
%or
% \tkzDefCircle[ex](B,C,A)
% \tkzGetFirstPoint{J_c}
% \tkzGetSecondPoint{Tc}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawPoints(A,B,C,Tc,J_c)
\tkzDrawCircle[red](J_c,Tc)
\tkzDrawLines[add=1.5 and 0](A,C B,C)
\tkzLabelPoints(A,B,C,Tc,J_c)
\end{tikzpicture}
```

4.6 euler 选项：欧拉点

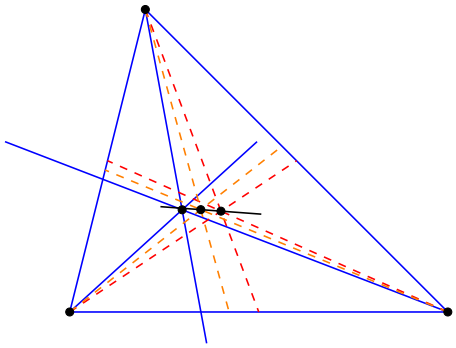
欧拉点是欧拉圆的圆心，欧拉圆又称九点圆或费尔巴哈圆的圆心。欧拉圆是通过三角形 ABC 三个顶点向对边作垂线形成的三个垂脚 H_A、H_B 和 H_C 的圆，欧拉在 1765 年证明该圆同时通过三角形 ABC 三个边的中点 M_A、M_B 和 M_C。根据费尔巴哈定理，欧拉圆也通过三角形 ABC 三个顶点与重心 H 连线线段的中点 E_A、E_B 和 E_C。(https://mathworld.wolfram.com/Nine-PointCircle.html)




```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefSpcTriangle[medial,name=M](A,B,C){_A,_B,_C}
\tkzDefTriangleCenter[euler](A,B,C)
\tkzGetPoint{N} % I= N nine points
\tkzDefTriangleCenter[ortho](A,B,C)
\tkzGetPoint{H}
\tkzDefMidPoint(A,H) \tkzGetPoint{E_A}
\tkzDefMidPoint(C,H) \tkzGetPoint{E_C}
\tkzDefMidPoint(B,H) \tkzGetPoint{E_B}
\tkzDefSpcTriangle[ortho,name=H](A,B,C){_A,_B,_C}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawCircle(N,E_A)
\tkzDrawSegments[blue](A,H_A B,H_B C,H_C)
\tkzDrawPoints(A,B,C,N,H)
\tkzDrawPoints[red](M_A,M_B,M_C)
\tkzDrawPoints[blue]( H_A,H_B,H_C)
\tkzDrawPoints[green](E_A,E_B,E_C)
\tkzAutoLabelPoints[center=N, font=\scriptsize]%
(A,B,C,M_A,M_B,M_C,H_A,H_B,H_C,E_A,E_B,E_C)
\tkzLabelPoints[font=\scriptsize](H,N)
\tkzMarkSegments[mark=s|,size=3pt,
color=blue,line width=1pt](B,E_B E_B,H)
\end{tikzpicture}
```

4.7 symmedian 选项：陪位重心

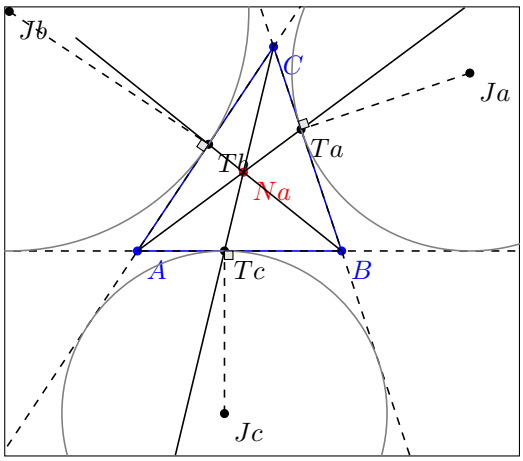
设 AN、BM、CE 是三角形 ABC 的三条中线，N'、M'、E' 分别是三条边 BC、CA、AB 上的点，若 $\widehat{BAN'}=\widehat{NAC}$、$\widehat{CBM}=\widehat{M'BA}$、$\widehat{ACE}=\widehat{E'CB}$，则三条直线 AN'、BM'、CE' 交于一点 K，该点称为陪位重心。



```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(5,0){B}
\tkzDefPoint(1,4){C}
\tkzDefTriangleCenter[symmedian](A,B,C) \tkzGetPoint{K}
\tkzDefTriangleCenter[median](A,B,C)    \tkzGetPoint{G}
\tkzDefTriangleCenter[in](A,B,C)        \tkzGetPoint{I}
\tkzDefSpcTriangle[centroid,name=M](A,B,C){a,b,c}
\tkzDefSpcTriangle[incentral,name=I](A,B,C){a,b,c}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawLines[add = 0 and 2/3,blue](A,K B,K C,K)
\tkzDrawSegments[red,dashed](A,Ma B,Mb C,Mc)
\tkzDrawSegments[orange,dashed](A,Ia B,Ib C,Ic)
\tkzDrawLine[add=2 and 2](G,I)
\tkzDrawPoints(A,B,C,K,G,I)
\end{tikzpicture}
```

4.8 nagel 选项：界心 (Nagel 点)

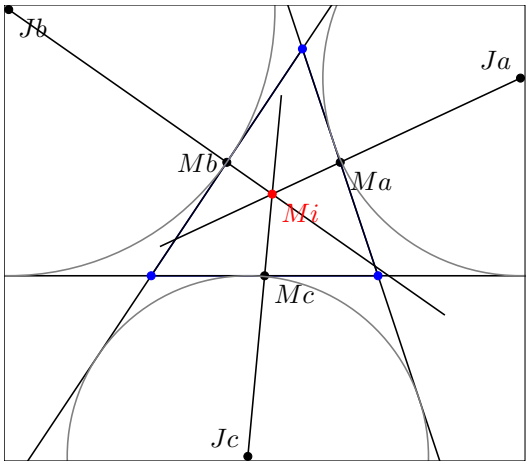
令 Ta、Tb 和 Tc 分别为旁切圆与三角形三条边的切点，连线 ATa、BTb 和 CTc，其交点称为 Nagel 点，俗称三角形的“界心”。Weisstein, Eric W. “Nagel point”. From MathWorld–A Wolfram Web Resource.




```
\begin{tikzpicture}[scale=0.45]
\tkzDefPoints{0/0/A,6/0/B,4/6/C}
\tkzDefSpcTriangle[ex](A,B,C){Ja,Jb,Jc}
\tkzDefSpcTriangle[extouch](A,B,C){Ta,Tb,Tc}
\tkzDrawPoints(Ja,Jb,Jc,Ta,Tb,Tc)
\tkzLabelPoints(Ja,Jb,Jc,Ta,Tb,Tc)
\tkzDrawPolygon[blue](A,B,C)
\tkzDefTriangleCenter[nagel](A,B,C) \tkzGetPoint{Na}
\tkzDrawPoints[blue](B,C,A)
\tkzDrawPoints[red](Na)
\tkzLabelPoints[blue](B,C,A)
\tkzLabelPoints[red](Na)
\tkzDrawLines[add=0 and 1](A,Ta B,Tb C,Tc)
\tkzShowBB \tkzClipBB
\tkzDrawLines[add=1 and 1,dashed](A,B B,C C,A)
\tkzDrawCircles[ex,gray](A,B,C C,A,B B,C,A)
\tkzDrawSegments[dashed](Ja,Ta Jb,Tb Jc,Tc)
\tkzMarkRightAngles[fill=gray!20](Ja,Ta,C %
Jb,Tb,A Jc,Tc,B)
\end{tikzpicture}
```

4.9 mittenpunkt 选项

三个旁切圆圆心与对应边中点连线的交点。




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoints{0/0/A,6/0/B,4/6/C}
\tkzDefSpcTriangle[centroid](A,B,C){Ma,Mb,Mc}
\tkzDefSpcTriangle[ex](A,B,C){Ja,Jb,Jc}
\tkzDefSpcTriangle[extouch](A,B,C){Ta,Tb,Tc}
\tkzDefTriangleCenter[mittenpunkt](A,B,C)
\tkzGetPoint{Mi}
\tkzDrawPoints(Ma,Mb,Mc,Ja,Jb,Jc)
\tkzClipBB
\tkzDrawPolygon[blue](A,B,C)
\tkzDrawLines[add=0 and 1](Ja,Ma %
Jb,Mb Jc,Mc)
\tkzDrawLines[add=1 and 1](A,B A,C B,C)
\tkzDrawCircles[gray](Ja,Ta Jb,Tb Jc,Tc)
\tkzDrawPoints[blue](B,C,A)
\tkzDrawPoints[red](Mi)
\tkzLabelPoints[red](Mi)
\tkzLabelPoints[left](Mb)
\tkzLabelPoints(Ma,Mc,Jb,Jc)
\tkzLabelPoints[above left](Ja,Jc)
\tkzShowBB
\end{tikzpicture}
```

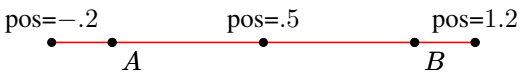
第五章 点的定义 — 直线或圆上的点

5.1 定义直线上的点

\tkzDefPointOnLine[选项]($pt1, pt2$)

参数	默认值	含义
$pt1, pt2$	无	直线上的两个点
选项	**默认值**	**含义**
pos=nb	无	距离起点 A 相对于 AB 线段长的比例

示例：

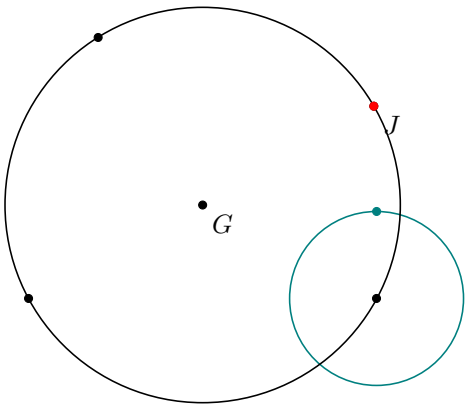



```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,4/0/B}
\tkzDrawLine[red](A,B)
\tkzDefPointOnLine[pos=1.2](A,B)
\tkzGetPoint{P}
\tkzDefPointOnLine[pos=-0.2](A,B)
\tkzGetPoint{R}
\tkzDefPointOnLine[pos=0.5](A,B)
\tkzGetPoint{S}
\tkzDrawPoints(A,B,P)
\tkzLabelPoints(A,B)
\tkzLabelPoint[above](P){pos=$1.2$}
\tkzLabelPoint[above](R){pos=$-.2$}
\tkzLabelPoint[above](S){pos=$.5$}
\tkzDrawPoints(A,B,P,R,S)
\tkzLabelPoints(A,B)
\end{tikzpicture}
```

5.2 定义圆上的点

\tkzDefPointOnCircle[选项])

选项	默认值	含义
angle	0	与横轴夹角
center	\tkzPointResult	圆心
radius	\tkzLengthResult	半径




```
\begin{tikzpicture}[scale=1.15]
\tkzDefPoints{0/0/A,4/0/B,0.8/3/C}
\tkzDefPointOnCircle[angle=90,center=B,radius=1 cm]
\tkzGetPoint{I}
\tkzDefCircle[circum](A,B,C)
\tkzGetPoint{G} \tkzGetLength{rG}
\tkzDefPointOnCircle[angle=30,center=G,radius=\rG pt]
\tkzGetPoint{J}
\tkzDrawCircle[R,teal](B,1cm)
\tkzDrawPoint[teal](I)
\tkzDrawPoints(A,B,C)
\tkzDrawCircle(G,J)
\tkzDrawPoints(G,J)
\tkzDrawPoint[red](J)
\tkzLabelPoints(G,J)
\end{tikzpicture}
```

第六章 点的定义 — 利用坐标变换定义点

这些变换主要有：

- 平移;
- 缩放;
- 轴对称;
- 中心对称;
- 正交投影;
- 旋转 (度或弧度);
- 相对于圆的反转.

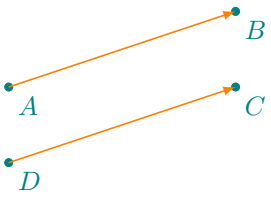
\tkzDefPointBy[选项](*pt*)

参数是一个已知点，结果存储于\tkzPointResult 命令
可用\tkzGetPoint 命令保存该点，并为点命名

参数	含义	样例
pt	一个已知点的名称	(*A*)
选项		**样例**
translation(平移)	= from #1 to #2	[translation=from A to B](E)
projection(投影)	= onto #1-- #2	[projection=onto A–B](E)
symmetry(中心对称)	= center #1	[symmetry=center A](E)
rotation in rad(旋转弧度)	= center #1 angle #2	[rotation in rad=center O angle pi/3](E)
homothety(位拟)	= center #1 ratio #2	[homothety=center A ratio .5](E)
reflection(轴对称)	= over #1-- #2	[reflection=over A–B](E)
rotation(旋转角度)	= center #1 angle #2	[rotation=center O angle 30](E)
inversion(反转)	= center #1 through #2	[inversion =center O through A](E)

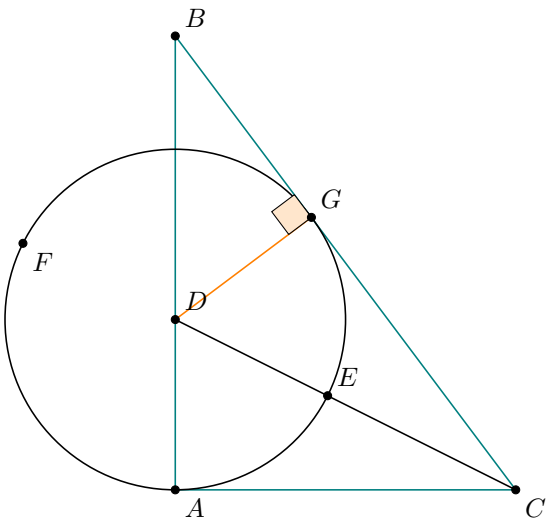
该命令仅定义一个点，并不绘制该点

6.1 translation-平移示例



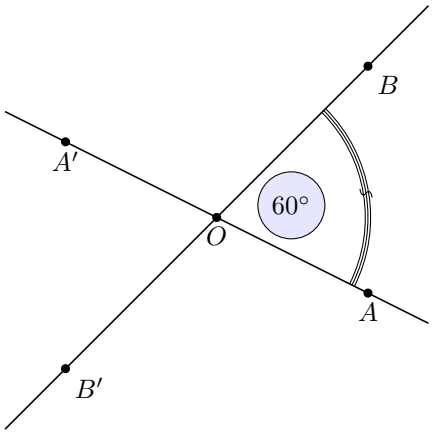

```
\begin{tikzpicture}[>=latex]
\tkzDefPoints{0/0/A,3/1/B,3/0/C}
\tkzDefPointBy[translation= from B to A](C)
\tkzGetPoint{D}
\tkzDrawPoints[teal](A,B,C,D)
\tkzLabelPoints[color=teal](A,B,C,D)
\tkzDrawSegments[orange,->](A,B D,C)
\end{tikzpicture}
```

6.2 projection-投影示例



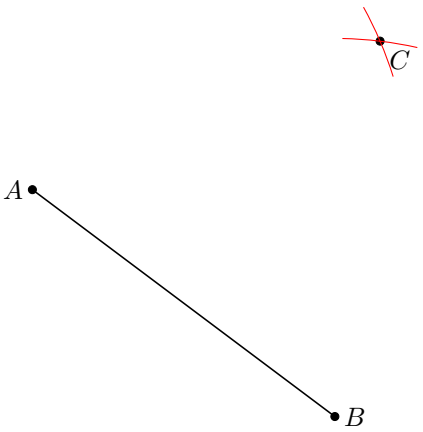

```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){A} \tkzDefPoint(0,4){B}
\tkzDefTriangle[pythagore](B,A) \tkzGetPoint{C}
\tkzDefLine[bisector](B,C,A)    \tkzGetPoint{c}
\tkzInterLL(C,c)(A,B)           \tkzGetPoint{D}
\tkzDefPointBy[projection=onto B-C](D)
\tkzGetPoint{G}
\tkzInterLC(C,D)(D,A) \tkzGetPoints{E}{F}
\tkzDrawPolygon[teal](A,B,C)
\tkzDrawSegment(C,D) \tkzDrawCircle(D,A)
\tkzDrawSegment[orange](D,G)
\tkzMarkRightAngle[fill=orange!20](D,G,B)
\tkzDrawPoints(A,C,F,B,D,E,G)
\tkzLabelPoints(A,C,F) \tkzLabelPoints[above right](B,D,E,G)
\end{tikzpicture}
```

6.3 symmetry-中心对称示例



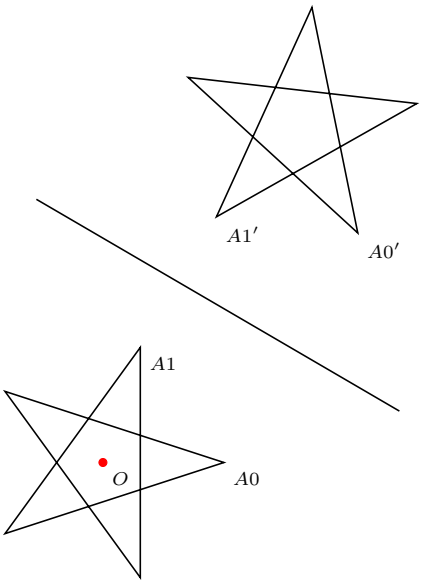

```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/O,2/-1/A,2/2/B}
% 注意：下条命令末尾的 {} 不可省略
\tkzDefPointsBy[symmetry=center O](B,A){}
\tkzDrawLines(A,A' B,B')
\tkzMarkAngle[mark=s,arc=lll,size=2 cm,mkcolor=red](A,O,B)
\tkzLabelAngle[pos=1,circle,draw,fill=blue!10](A,O,B){$60^{\circ}$}
\tkzDrawPoints(A,B,O,A',B')
\tkzLabelPoints(B,B')
\tkzLabelPoints[below](A,A',O)
\end{tikzpicture}
```

6.4 rotation in rad-旋转弧度示例



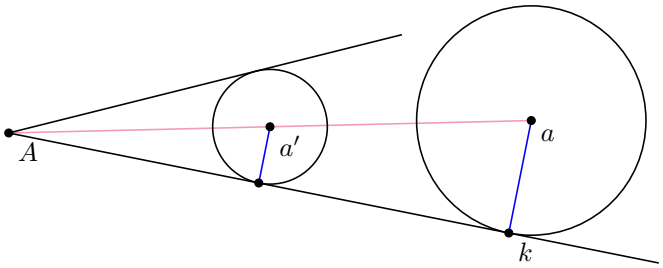

```
\begin{tikzpicture}
\tkzDefPoint["$A$" left](1,5){A}
\tkzDefPoint["$B$" right](5,2){B}
\tkzDefPointBy[rotation in rad=center A angle pi/3](B)
\tkzGetPoint{C}
\tkzDrawSegment(A,B)
\tkzDrawPoints(A,B,C)
\tkzCompass[color=red](A,C)
\tkzCompass[color=red](B,C)
\tkzLabelPoints(C)
\end{tikzpicture}
```

6.5 rotation-旋转角度,reflection-轴对称示例



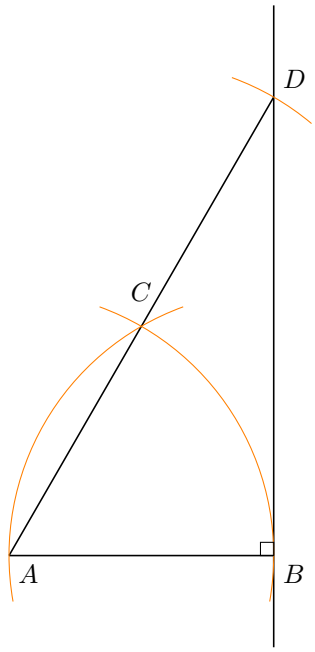

```
\begin{tikzpicture}[scale=.8]
\tkzDefPoints{1.5/-1.5/C,-4.5/2/D,-4/-2/O,-2/-2/A}
\foreach \i in {0,1,...,4}{%
\pgfmathparse{0+\i * 72}
\tkzDefPointBy[rotation=%
center O angle \pgfmathresult](A)
\tkzGetPoint{A\i}
\tkzDefPointBy[reflection = over C-D](A\i)
\tkzGetPoint{A\i'}}
\tkzDrawPolygon(A0, A2, A4, A1, A3)
\tkzDrawPolygon(A0', A2', A4', A1', A3')
\tkzDrawPoint[red](O)
\tkzLabelPoints[font=\scriptsize](A0,A1,A0',A1',O)
\tkzDrawLine[add= -0.1 and .1](C,D)
\end{tikzpicture}
```

6.6 homothety-位拟和 projection-投影示例



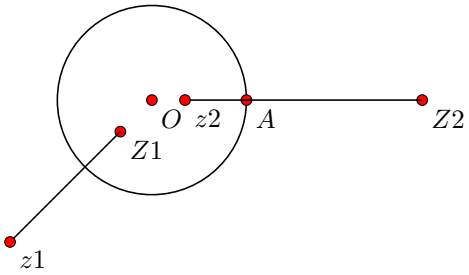

```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,1){A}   \tkzDefPoint(5,0){B}    \tkzDefPoint(4,2){C}
\tkzDefLine[bisector](B,A,C)                   \tkzGetPoint{a}
\tkzDrawLine[add=0 and 0,color=magenta!50 ](A,a)
\tkzDefPointBy[homothety=center A ratio .5](a) \tkzGetPoint{a'}
\tkzDefPointBy[projection = onto A-B](a')      \tkzGetPoint{k'}
\tkzDefPointBy[projection = onto A-B](a)       \tkzGetPoint{k}
\tkzDrawLines[add= 0 and .3](A,k A,C)
\tkzDrawSegments[blue](a',k' a,k)
\tkzDrawPoints(a,a',k,k',A)
\tkzDrawCircles(a',k' a,k)
\tkzLabelPoints(a,a',k,A)
\end{tikzpicture}
```

6.7 symmetry 中心对称,-rotation-旋转角度示例



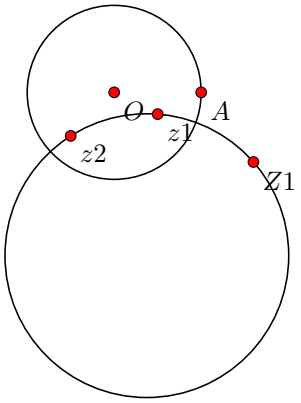

```
\begin{tikzpicture}[scale=0.7]
\tkzDefPoint(0,0){A} \tkzDefPoint(5,0){B}
\tkzDrawSegment(A,B)
\tkzDefPointBy[rotation=center A angle 60](B)
\tkzGetPoint{C}
\tkzDefPointBy[symmetry=center C](A)
\tkzGetPoint{D}
\tkzDrawSegment(A,tkzPointResult)
\tkzDrawLine(B,D)
\tkzDrawArc[orange,delta=10](A,B)(C)
\tkzDrawArc[orange,delta=10](B,C)(A)
\tkzDrawArc[orange,delta=10](C,D)(D)
\tkzMarkRightAngle(D,B,A)
\tkzLabelPoints(A,B)
\tkzLabelPoints[above=6pt](C)
\tkzLabelPoints[above right](D)
\end{tikzpicture}
```

6.8 inversion-反转示例




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){O}
\tkzDefPoint(1,0){A}
\tkzDefPoint(-1.5,-1.5){z1}
\tkzDefPoint(0.35,0){z2}
\tkzDefPointBy[inversion = center O through A](z1)
\tkzGetPoint{Z1}
\tkzDefPointBy[inversion = center O through A](z2)
\tkzGetPoint{Z2}
\tkzDrawCircle(O,A)
\tkzDrawPoints[color=black,fill=red,size=4](A,Z1,Z2)
\tkzDrawSegments(z1,Z1 z2,Z2)
\tkzDrawPoints[color=black,fill=red,size=4](O,z1,z2)
\tkzLabelPoints(O,A,z1,z2,Z1,Z2)
\end{tikzpicture}
```

6.9 点的反转：正交圆




```
\begin{tikzpicture}[scale=1.15]
\tkzDefPoint(0,0){O} \tkzDefPoint(1,0){A}
\tkzDrawCircle(O,A)
\tkzDefPoint(0.5,-0.25){z1} \tkzDefPoint(-0.5,-0.5){z2}
\tkzDefPointBy[inversion = center O through A](z1)
\tkzGetPoint{Z1}
\tkzCircumCenter(z1,z2,Z1) \tkzGetPoint{c}
\tkzDrawCircle(c,Z1)
\tkzDrawPoints[color=black,fill=red,size=4](z1,z2,Z1,O,A)
\tkzLabelPoints(z1,z2,Z1,O,A)
\end{tikzpicture}
```

6.10 通过变换定义多个点

\tkzDefPointsBy 命令是单点变换命令的变体，用于定义多点变换。必须在圆括号中，通过参数指定变换点名称，也可以在大括号中给出变换后点的名称。

```
\tkzDefPointsBy[translation= from A to A'](B,C){}
```

变换后的点是 B' 和 C'.

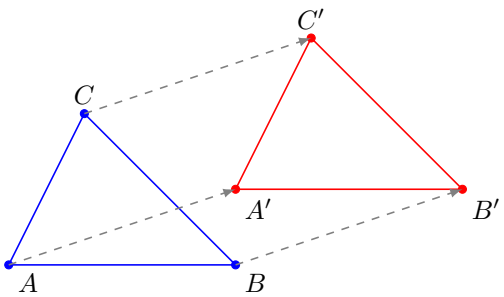
```
\tkzDefPointsBy[translation= from A to A'](B,C){D,E}
```

变换后的点是 D 和 E.

```
\tkzDefPointsBy[translation= from A to A'](B)
```

变换后的点是 B'

6.11 变换多点示例




```
\begin{tikzpicture}[>=latex]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,1){A'}
\tkzDefPoint(3,0){B}
\tkzDefPoint(1,2){C}
\tkzDefPointsBy[translation= from A to A'](B,C){}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawPolygon[color=red](A',B',C')
\tkzDrawPoints[color=blue](A,B,C)
\tkzDrawPoints[color=red](A',B',C')
\tkzLabelPoints(A,B,A',B')
\tkzLabelPoints[above](C,C')
\tkzDrawSegments[color = gray,->,%
style=dashed](A,A' B,B' C,C')
\end{tikzpicture}
```

第七章 点的定义 — 向量点

\tkzDefPointWith 命令可通过多种方案定义满足特定向量条件的点，此时，需要用两个点作为参数，也就是一个向量。不同的选项用于设置通过共线向量或正交向量的方式定义新点，向量的长度可以与第 1 个向量的长度成正比，也可以与单位向量成正比。如果该点仅做临时使用，则不需要立即命名，使用\tkzPointResult 命令即可。也可使用\tkzGetPoint 命令保存该点，并为其命名。

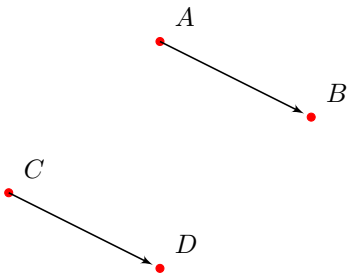
可以通过选项设置指定点与所求点之间的距离，通常，该距离是参数中给定 2 个点之间的距离，如果使用了 “normed” 选项，则定义的点的距离为 1 cm。然后可以通过比例系数 K 选项对其进行缩放。

\tkzDefPointWith[选项]($pt1, pt2$)

满足向量条件的点的定义

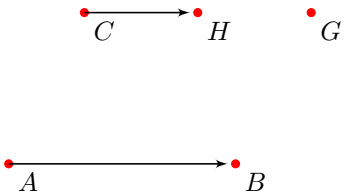
参数	含义	说明
$pt1, pt2$	点对	结果保存于\tkzPointResult 命令
选项	**样例**	**说明** (现假定由\tkzGetPoint{C} 得到该点)
colinear= at #1	[colinear= at C](A,B)	$\overrightarrow{CD} = \overrightarrow{AB}$
colinear normed= at #1	[colinear normed= at C](A,B)	$\overrightarrow{CD} = \overrightarrow{AB}$
orthogonal	[orthogonal](A,B)	$AC = AB$ 和 $\overrightarrow{AC} \perp \overrightarrow{AB}$
orthogonal normed	[orthogonal normed](A,B)	$AC = 1$ 和 $\overrightarrow{AC} \perp \overrightarrow{AB}$
linear	[linear](A,B)	$\overrightarrow{AC} = K \times \overrightarrow{AB}$
linear normed	[linear normed](A,B)	$AC = K$ 和 $\overrightarrow{AC} = k \times \overrightarrow{AB}$
K	[linear](A,B),K=2	$\overrightarrow{AC} = 2 \times \overrightarrow{AB}$

7.1 colinear at 选项示例



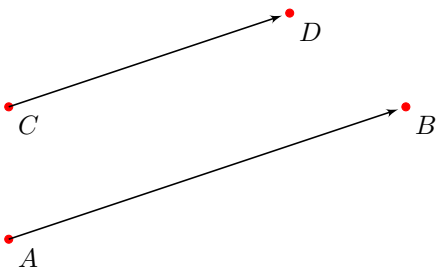

```
\begin{tikzpicture}[scale=1]
\tikzset{vect/.style={->, shorten >=3pt,>=latex'}}
\tkzDefPoints{2/3/A,4/2/B,0/1/C}
\tkzDefPointWith[colinear=at C](A,B)
\tkzGetPoint{D}
\tkzDrawPoints[color=red](A,B,C,D)
\tkzLabelPoints[above right=3pt](A,B,C,D)
\tkzDrawSegments[vect](A,B C,D)
\end{tikzpicture}
```

7.2 colinear at 带 K 选项示例




```
\begin{tikzpicture}[vect/.style={->, shorten >=3pt,>=latex'}]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefPoint(1,2){C}
\tkzDefPointWith[colinear=at C](A,B) \tkzGetPoint{G}
\tkzDefPointWith[colinear=at C,K=0.5](A,B)
\tkzGetPoint{H}
\tkzLabelPoints(A,B,C,G,H)
\tkzDrawPoints[red](A,B,C,G,H)
\tkzDrawSegments[vect](A,B C,H)
\end{tikzpicture}
```

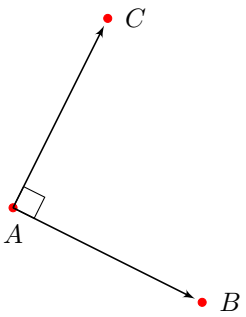
7.3 colinear at 带 $K = \frac{\sqrt{2}}{2}$ 选项示例




```
\begin{tikzpicture}[vect/.style={->, shorten >=3pt,
>=latex'}, scale=1.75]
\tkzDefPoint(1,1){A}
\tkzDefPoint(4,2){B}
\tkzDefPoint(1,2){C}
\tkzDefPointWith[colinear=at C,K=sqrt(2)/2](A,B)
\tkzGetPoint{D}
\tkzDrawPoints[color=red](A,B,C,D)
\tkzLabelPoints(A,B,C,D)
\tkzDrawSegments[vect](A,B C,D)
\end{tikzpicture}
```

7.4 orthogonal 选项示例

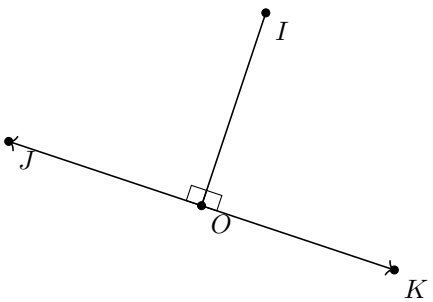
因 $K = 1$，所以 $AB = AC$。




```
\begin{tikzpicture}[vect/.style={->,shorten >=3pt,
>=latex'},scale=1.25]
\tkzDefPoint(2,3){A}  \tkzDefPoint(4,2){B}
\tkzDefPointWith[orthogonal,K=1](A,B)
\tkzGetPoint{C}
\tkzDrawPoints[color=red](A,B,C)
\tkzLabelPoints[right=3pt](B,C)
\tkzLabelPoints[below=3pt](A)
\tkzDrawSegments[vect](A,B A,C)
\tkzMarkRightAngle(B,A,C)
\end{tikzpicture}
```

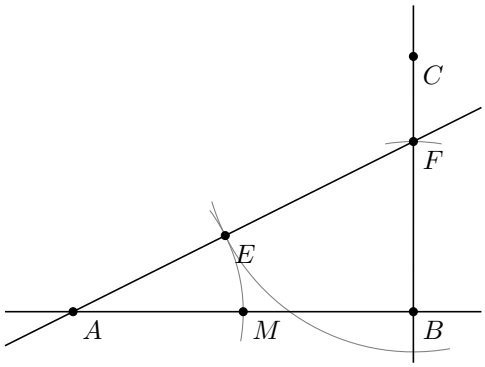
7.5 orthogonal 带 $K = -1$ 选项示例

因 $|K| = 1$，所以 $OI = OJ = OK$。



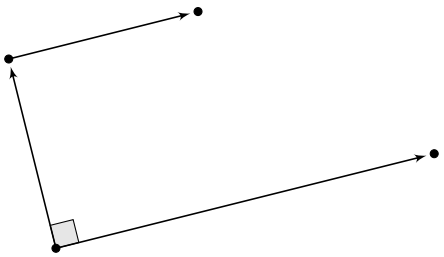

```
\begin{tikzpicture}[scale=0.85]
\tkzDefPoint(1,2){O}  \tkzDefPoint(2,5){I}
\tkzDefPointWith[orthogonal](O,I)
\tkzGetPoint{J}
\tkzDefPointWith[orthogonal,K=-1](O,I)
\tkzGetPoint{K}
\tkzDrawSegment(O,I)
\tkzDrawSegments[->](O,J O,K)
\tkzMarkRightAngles(I,O,J I,O,K)
\tkzDrawPoints(O,I,J,K)
\tkzLabelPoints(O,I,J,K)
\end{tikzpicture}
```

7.6 orthogonal 选项综合示例




```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoints{0/0/A,6/0/B}
\tkzDefMidPoint(A,B)
\tkzGetPoint{I}
\tkzDefPointWith[orthogonal,K=-.75](B,A)
\tkzGetPoint{C}
\tkzInterLC(B,C)(B,I)
\tkzGetPoints{D}{F}
\tkzDuplicateSegment(B,F)(A,F)
\tkzGetPoint{E}
\tkzDrawArc[delta=10](F,E)(B)
\tkzInterLC(A,B)(A,E)
\tkzGetPoints{N}{M}
\tkzDrawArc[delta=10](A,M)(E)
\tkzDrawLines(A,B B,C A,F)
\tkzCompass(B,F)
\tkzDrawPoints(A,B,C,F,M,E)
\tkzLabelPoints(A,B,C,F,M,E)
\end{tikzpicture}
```

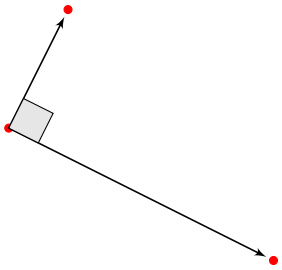
7.7 colinear 和 orthogonal 选项示例



```
\begin{tikzpicture}[vect/.style={->,shorten >=3pt,
>=latex'}, scale=1.25]
\tkzDefPoint(2,1){A}
\tkzDefPoint(6,2){B}
\tkzDefPointWith[orthogonal,K=.5](A,B)
\tkzGetPoint{C}
\tkzDefPointWith[colinear=at C,K=.5](A,B)
\tkzGetPoint{D}
\tkzMarkRightAngle[fill=gray!20](B,A,C)
\tkzDrawSegments[vect](A,B A,C C,D)
\tkzDrawPoints(A,...,D)
\end{tikzpicture}
```

7.8 orthogonal normed 带 $K = 1$ 选项示例

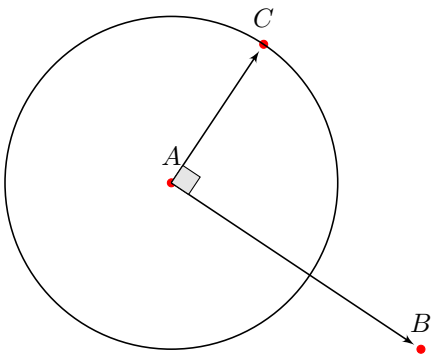
$AC = 1$.



```
\begin{tikzpicture}[vect/.style={->,shorten >=3pt,
>=latex'},scale=1.75]
\tkzDefPoint(2,3){A}
\tkzDefPoint(4,2){B}
\tkzDefPointWith[orthogonal normed](A,B)
\tkzGetPoint{C}
\tkzDrawPoints[color=red](A,B,C)
\tkzDrawSegments[vect](A,B A,C)
\tkzMarkRightAngle[fill=gray!20](B,A,C)
\end{tikzpicture}
```

7.9 orthogonal normed 和 $K = 2$ 选项示例

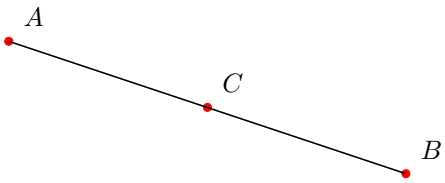
因 $K = 2$，所以 $AC = 2$。




```
\begin{tikzpicture}[vect/.style={->,shorten >=3pt,
>=latex'}, scale=1.10]
\tkzDefPoint(2,3){A}
\tkzDefPoint(5,1){B}
\tkzDefPointWith[orthogonal normed,K=2](A,B)
\tkzGetPoint{C}
\tkzDrawPoints[color=red](A,B,C)
\tkzDrawCircle[R](A,2cm)
\tkzDrawSegments[vect](A,B A,C)
\tkzMarkRightAngle[fill=gray!20](B,A,C)
\tkzLabelPoints[above=3pt](A,B,C)
\end{tikzpicture}
```

7.10 linear 选项示例

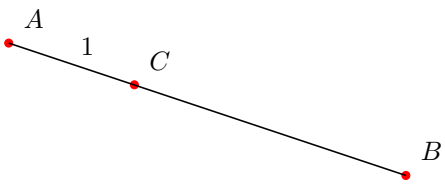
在此，取 $K = 0.5$。
这相当于给一个向量乘了一个实数，本例中是 $[AB]$ 的中点。




```
\begin{tikzpicture}[scale=1.75]
\tkzDefPoint(1,3){A}
\tkzDefPoint(4,2){B}
\tkzDefPointWith[linear,K=0.5](A,B)
\tkzGetPoint{C}
\tkzDrawPoints[color=red](A,B,C)
\tkzDrawSegment(A,B)
\tkzLabelPoints[above right=3pt](A,B,C)
\end{tikzpicture}
```

7.11 linear normed 选项示例

在下面的实例中，$AC = 1$ 并且 C 属于 (AB)。




```
\begin{tikzpicture}[scale=1.75]
\tkzDefPoint(1,3){A}
\tkzDefPoint(4,2){B}
\tkzDefPointWith[linear normed](A,B)
\tkzGetPoint{C}
\tkzDrawPoints[color=red](A,B,C)
\tkzDrawSegment(A,B)
\tkzLabelSegment(A,C){$1$}
\tkzLabelPoints[above right=3pt](A,B,C)
\end{tikzpicture}
```

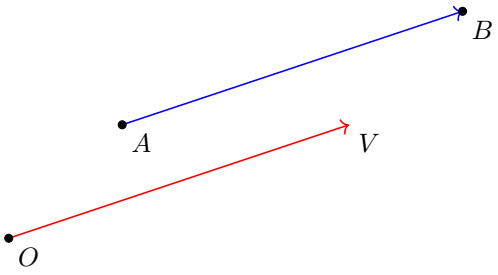
7.12 获取向量坐标分量

\tkzGetVectxy$(A,B)\{text\}$

获得一个向量的坐标分量

参数	样例	说明
$(pt1,pt2)$\{前缀字母\}	\tzkGetVectzy$(A,B)$\{V\}	\Vx,\Vy 是向量 $\overrightarrow{AB}$ 的坐标分量

7.13 使用\tkzGetVectxy 命令实现坐标变换




```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){O}
\tkzDefPoint(1,1){A}
\tkzDefPoint(4,2){B}
\tkzGetVectxy(A,B){v}
\tkzDefPoint(\vx,\vy){V}
\tkzDrawSegment[->,color=red](O,V)
\tkzDrawSegment[->,color=blue](A,B)
\tkzDrawPoints(A,B,O)
\tkzLabelPoints(A,B,O,V)
\end{tikzpicture}
```

第八章 定义随机点

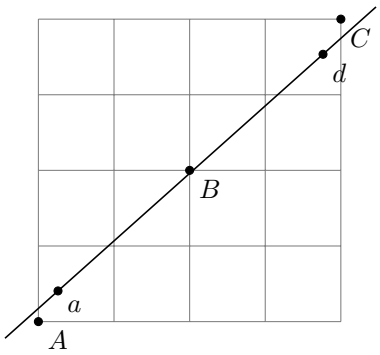
可以使用以下四种方式定义随机点

1. 矩形内的点;
2. 线段上的点;
3. 直线上的点;
4. 圆上的点;

\tkzDefRandPointOn[命令选项]

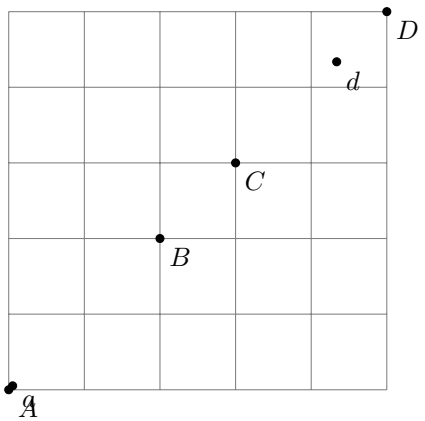
命令选项	举例	含义
rectangle=pt1 and pt2	[rectangle=A and B]	矩形内的随机点
segment= pt1–pt2	[segment=A–B]	线段上的随机点
line=pt1–pt2	[line=A–B]	直线上的随机点
circle =center pt1 radius dim	[circle = center A radius 2 cm]	圆上的随机点
circle through=center pt1 through pt2	[circle through= center A through B]	圆上的随机点
disk through=center pt1 through pt2	[disk through=center A through B]	圆内的随机点

8.1 矩形内的随机点



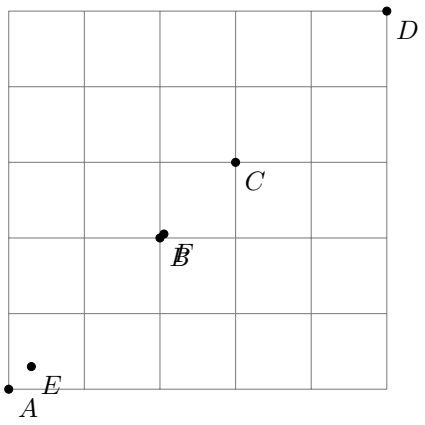

```
\begin{tikzpicture}
\tkzInit[xmax=4,ymax=4]
\tkzGrid
\tkzDefPoints{0/0/A,2/2/B,4/4/C}
\tkzDefRandPointOn[rectangle = A and B]
\tkzGetPoint{a}
\tkzDefRandPointOn[rectangle = B and C]
\tkzGetPoint{d}
\tkzDrawLine(a,d)
\tkzDrawPoints(A,B,C,a,d)
\tkzLabelPoints(A,B,C,a,d)
\end{tikzpicture}
```

8.2 线段上的随机点



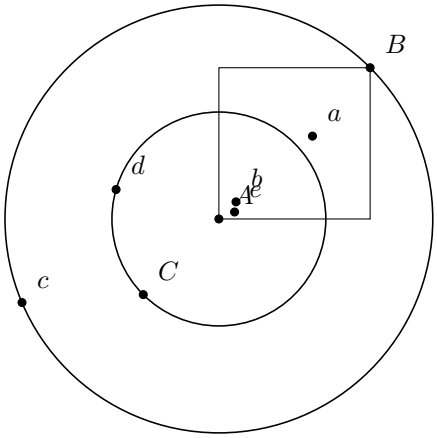

```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=5]
\tkzGrid
\tkzDefPoints{0/0/A,2/2/B,3/3/C,5/5/D}
\tkzDefRandPointOn[segment = A-B]\tkzGetPoint{a}
\tkzDefRandPointOn[segment = C-D]\tkzGetPoint{d}
\tkzDrawPoints(A,B,C,D,a,d)
\tkzLabelPoints(A,B,C,D,a,d)
\end{tikzpicture}
```

8.3 直线上的随机点



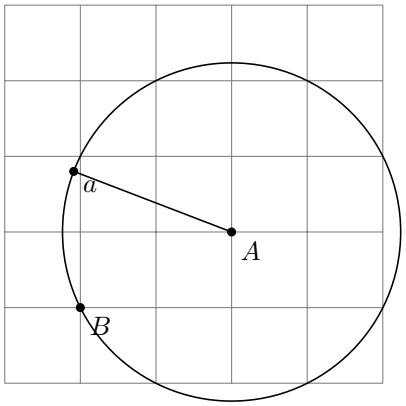

```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=5]
\tkzGrid
\tkzDefPoints{0/0/A,2/2/B,3/3/C,5/5/D}
\tkzDefRandPointOn[line = A-B]\tkzGetPoint{E}
\tkzDefRandPointOn[line = C-D]\tkzGetPoint{F}
\tkzDrawPoints(A,...,F)
\tkzLabelPoints(A,...,F)
\end{tikzpicture}
```

8.4 随机点综合示例



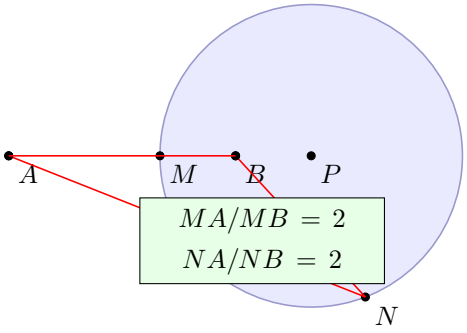

```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,2/2/B,-1/-1/C}
\tkzDefCircle[through=](A,C)
\tkzGetLength{rAC}
\tkzDrawCircle(A,C)
\tkzDrawCircle(A,B)
\tkzDefRandPointOn[rectangle=A and B]
\tkzGetPoint{a}
\tkzDefRandPointOn[segment=A-B]
\tkzGetPoint{b}
\tkzDefRandPointOn[circle=center A radius \rAC pt]
\tkzGetPoint{d}
\tkzDefRandPointOn[circle through= center A through B]
\tkzGetPoint{c}
\tkzDefRandPointOn[disk through=center A through B]
\tkzGetPoint{e}
\tkzLabelPoints[above right=3pt](A,B,C,a,b,...,e)
\tkzDrawPoints[](A,B,C,a,b,...,e)
\tkzDrawRectangle(A,B)
\end{tikzpicture}
```

8.5 圆上的随机点




```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=5]
\tkzGrid
\tkzDefPoints{3/2/A,1/1/B}
\tkzCalcLength[cm](A,B) \tkzGetLength{rAB}
\tkzDrawCircle[R](A,\rAB cm)
\tkzDefRandPointOn[circle = center A radius
\rAB cm] \tkzGetPoint{a}
\tkzDrawSegment(A,a)
\tkzDrawPoints(A,B,a)
\tkzLabelPoints(A,B,a)
\end{tikzpicture}
```

8.5.1 Apollonius 圆的随机示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/A,3/0/B}
\def\coeffK{2}
\tkzApolloniusCenter[K=\coeffK](A,B)
\tkzGetPoint{P}
\tkzDefApolloniusPoint[K=\coeffK](A,B)
\tkzGetPoint{M}
\tkzDefApolloniusRadius[K=\coeffK](A,B)
\tkzDrawCircle[R,color = blue!50!black, fill=blue!20,
opacity=.4](tkzPointResult,\tkzLengthResult pt)
\tkzDefRandPointOn[circle through= center P through M]
\tkzGetPoint{N}
\tkzDrawPoints(A,B,P,M,N)
\tkzLabelPoints(A,B,P,M,N)
\tkzDrawSegments[red](N,A N,B)
\tkzDrawPoints(A,B)
\tkzDrawSegments[red](A,B)
\tkzLabelCircle[R,draw,fill=green!10, text width=3cm,
text centered](P,\tkzLengthResult pt-20pt)(-120)
{$MA/MB=\coeffK$\\$NA/NB=\coeffK$}
\end{tikzpicture}
```

第九章 直线

在绘制直线之前，需要确定直线上的两个点，也可以通过垂直中分线、角平分线、平行线或垂线等方式确定特定直线

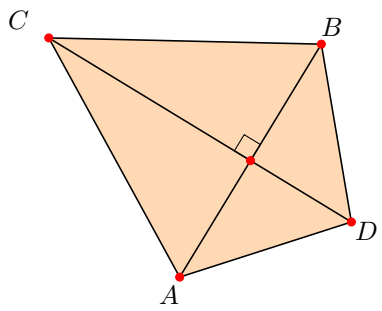
9.1 定义直线

\tkzDefLine[选项](*pt*1, *pt*2) or (*pt*1, *pt*2, *pt*3)

参数是 2 个或 3 个点的列表，根据具体问题，该命令得到 1 个或 2 个点

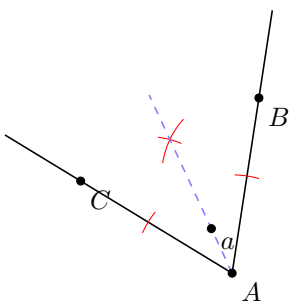
参数	样例	说明
(*pt*1, *pt*2)	(A,B)	
(*pt*1, *pt*2, *pt*3)	(A,B,C)	
选项	**默认**	**含义**
mediator		定义 2 个点连线的中垂线
perpendicular=through. . .	mediator	通过指定点的垂线
orthogonal=through. . .	mediator	同上
parallel=through. . .	mediator	通过指定点的平行线
bisector	mediator	内角角平分线
bisector out	mediator	外角角平分线
K	1	垂线的比例系数
normed	false	线段归一化

9.1.1 mediator 选项示例



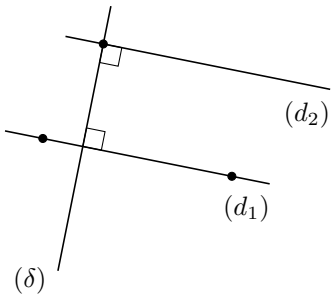

```
\begin{tikzpicture}[rotate=25]
\tkzDefPoints{-2/0/A,1/2/B}
\tkzDefLine[mediator](A,B)           \tkzGetPoints{C}{D}
\tkzDefPointWith[linear,K=.75](C,D) \tkzGetPoint{D}
\tkzDefMidPoint(A,B)                 \tkzGetPoint{I}
\tkzFillPolygon[color=orange!30](A,C,B,D)
\tkzDrawSegments(A,B C,D)
\tkzMarkRightAngle(B,I,C)
\tkzDrawSegments(D,B D,A C,B C,A)
\tkzDrawPoints[red](A,B,C,D,I)
\tkzAutoLabelPoints[center=I](A,B,C,D)
\end{tikzpicture}
```

9.1.2 bisector 和 normed 选项示例




```
\begin{tikzpicture}[rotate=25,scale=.65]
\tkzDefPoints{0/0/C, 2/-3/A, 4/0/B}
\tkzDefLine[bisector,normed](B,A,C) \tkzGetPoint{a}
\tkzDrawLines[add= 0 and .5](A,B A,C)
\tkzShowLine[bisector,gap=3,size=2,color=red](B,A,C)
\tkzDrawLines[blue!50,dashed,add= 0 and 3](A,a)
\tkzDrawPoints(A,B,C,a) \tkzLabelPoints(A,B,C,a)
\end{tikzpicture}
```

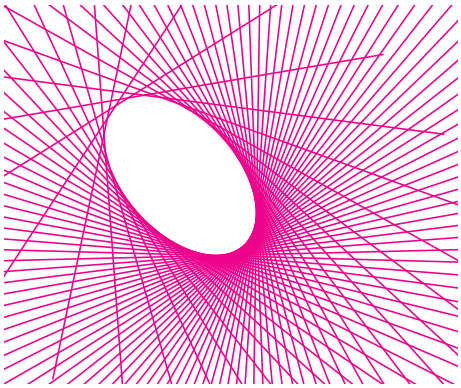
9.1.3 orthogonal 和 parallel 选项示例




```
\begin{tikzpicture}
\tkzDefPoints{-1.5/-0.25/A,1/-0.75/B,-0.7/1/C}
\tkzDrawLine(A,B)
\tkzLabelLine[pos=1.25,below left](A,B){$(d_1)$}
\tkzDrawPoints(A,B,C)
\tkzDefLine[orthogonal=through C](B,A) \tkzGetPoint{c}
\tkzDrawLine(C,c)
\tkzLabelLine[pos=1.25,left](C,c){$(\delta)$}
\tkzInterLL(A,B)(C,c) \tkzGetPoint{I}
\tkzMarkRightAngle(C,I,B)
\tkzDefLine[parallel=through C](A,B) \tkzGetPoint{c'}
\tkzDrawLine(C,c')
\tkzLabelLine[pos=1.25,below left](C,c'){$(d_2)$}
\tkzMarkRightAngle(I,C,c')
\end{tikzpicture}
```

9.1.4 循环画图

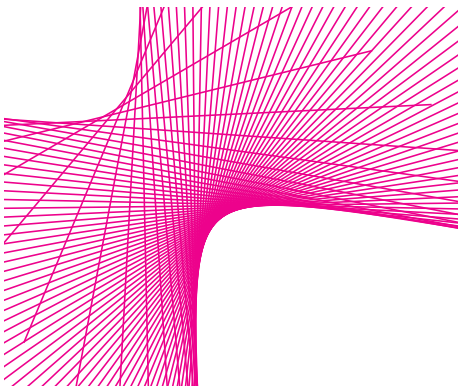
基于 D Rodriguez 用 pst-eucl 宏包绘制的 O. Reboux 设计的图形绘制。



```
\begin{tikzpicture}[scale=.5]
\tkzInit[xmin=-6,ymin=-4,xmax=6,ymax=6] % necessary
\tkzClip
\tkzDefPoint(0,0){O}
\tkzDefPoint(132:4){A}
\tkzDefPoint(5,0){B}
\foreach \ang in {5,10,...,360}{%
\tkzDefPoint(\ang:5){M}
\tkzDefLine[mediator](A,M)
\tkzDrawLine[color=magenta,add= 3 and %
3](tkzFirstPointResult,tkzSecondPointResult)}
\end{tikzpicture}
```

9.1.5 抛物线

基于 D Rodriguez 用 pst-eucl 宏包绘制的 O. Reboux 设计的图形绘制。
本例中，对定义的垂直平分线的两个端点进行命名是有必要的。



```
\begin{tikzpicture}[scale=.5]
\tkzInit[xmin=-6,ymin=-4,xmax=6,ymax=6]
\tkzClip
\tkzDefPoint(0,0){O}
\tkzDefPoint(132:5){A}
\tkzDefPoint(4,0){B}
\foreach \ang in {5,10,...,360}{%
\tkzDefPoint(\ang:4){M}
\tkzDefLine[mediator](A,M)
\tkzDrawLine[color=magenta,add= 3 and %
3](tkzFirstPointResult,tkzSecondPointResult)}
\end{tikzpicture}
```

9.2 定义圆的切线

圆的切线可以有两种，一是在圆上某点处的切线，这时已知圆上一点，通过命令定义出切线的另一个点。另一种是过圆外某点的圆的切线 (有两条切线)，这时已知圆外一点，通过命令定义出位于圆上的两个切线点

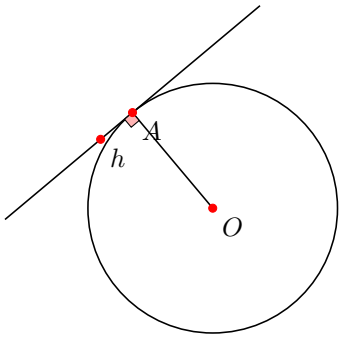
\tkzDefTangent[选项]($pt1, pt2$) or ($pt1, dim$)

$pt1$ 是圆心,$pt2$ 是圆上一点,dim 是半径

参数	样例	说明
($pt1, pt2$) or ($pt1, dim$)	(A,B) or (A,2cm)	
选项	**默认值**	**含义**
at=pt	at	圆上指定点的切线
from=pt	at	过圆外指定点的圆的切线
from with R=pt	at	同上，但圆需要通过圆心和半径参数指定

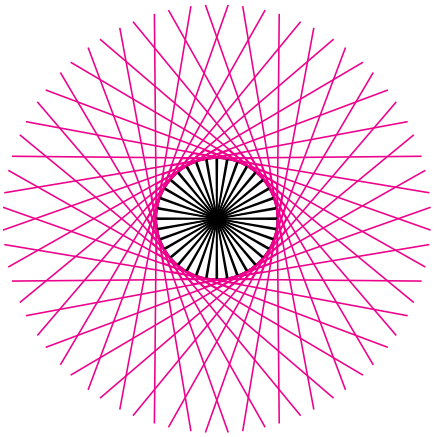
该命令仅定义切线的另一点，并不绘制切线。切线的第 2 个点可通过 *\tkzPointResult* 命令得到

9.2.1 圆上指定点的切线示例



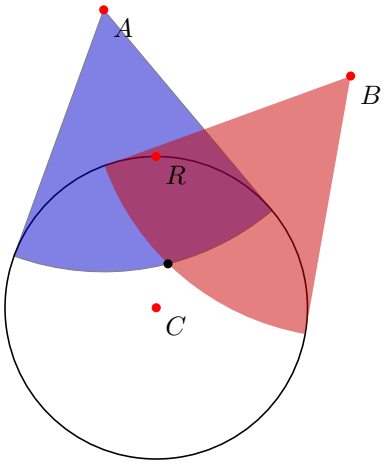

```
\begin{tikzpicture}[scale=.55]
\tkzDefPoint(0,0){O}
\tkzDefRandPointOn[circle=center O radius 3cm]
\tkzGetPoint{A} \tkzDrawSegment(O,A)
\tkzDrawCircle(O,A)
\tkzDefTangent[at=A](O)
\tkzGetPoint{h}
\tkzDrawLine[add = 4 and 3](A,h)
\tkzMarkRightAngle[fill=red!30](O,A,h)
\tkzDrawPoints[red](A,O,h)
\tkzLabelPoints(A,O,h)
\end{tikzpicture}
```

9.2.2 过圆外指定点的切线示例



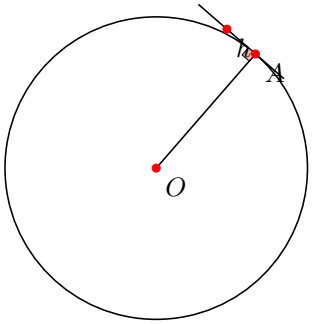
```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(3,3){c}
\tkzDefPoint(6,3){a0}
\tkzRadius=1 cm
\tkzDrawCircle[R](c,\tkzRadius)
\foreach \an in {0,10,...,350}{
\tkzDefPointBy[rotation=center c angle \an](a0)
\tkzGetPoint{a}
\tkzDefTangent[from with R = a](c,\tkzRadius)
\tkzGetPoints{e}{f}
\tkzDrawLines[color=magenta](a,f a,e)
\tkzDrawSegments(c,e c,f)
}%
\end{tikzpicture}
```

9.2.3 Andrew Mertz 示例



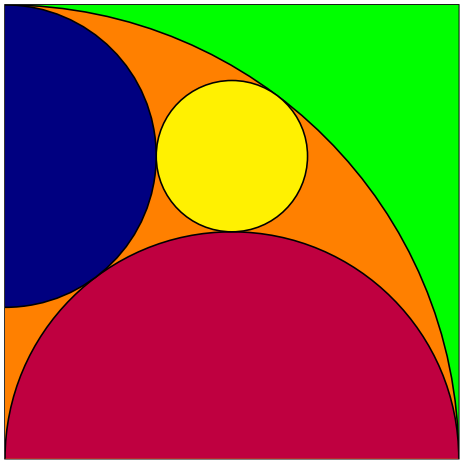

```
\begin{tikzpicture}[scale=0.5]
\tkzDefPoint(100:8){A} \tkzDefPoint(50:8){B}
\tkzDefPoint(0,0){C} \tkzDefPoint(0,4){R}
\tkzDrawCircle(C,R)
\tkzDefTangent[from = A](C,R)
\tkzGetPoints{D}{E}
\tkzDefTangent[from = B](C,R)
\tkzGetPoints{F}{G}
\tkzDrawSector[fill=blue!80!black,opacity=0.5](A,D)(E)
\tkzFillSector[color=red!80!black,opacity=0.5](B,F)(G)
\tkzInterCC(A,D)(B,F) \tkzGetSecondPoint{I}
\tkzDrawPoint[color=black](I)
\tkzDrawPoints[red](A,B,C,R)
\tkzLabelPoints(A,B,C,R)
\end{tikzpicture}
```

9.2.4 from with R 和 at 选项示例




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,0){O}
\tkzDefRandPointOn[circle=center O radius 4cm]
\tkzGetPoint{A}
\tkzDefTangent[at=A](O)
\tkzGetPoint{h}
\tkzDrawSegments(O,A)
\tkzDrawCircle(O,A)
\tkzDrawLine[add = 1 and 1](A,h)
\tkzMarkRightAngle[fill=red!30](O,A,h)
\tkzDrawPoints[red](O,A,h)
\tkzLabelPoints(O,A,h)
\end{tikzpicture}
```

9.2.5 from 选项示例



```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/B,0/8/A}
\tkzDefSquare(A,B)
\tkzGetPoints{C}{D}
\tkzDrawSquare(A,B)
\tkzClipPolygon(A,B,C,D)
\tkzDefPoints{4/8/F,4/0/E,4/4/Q}
\tkzFillPolygon[color = green](A,B,C,D)
\tkzDrawCircle[fill = orange](B,A)
\tkzDrawCircle[fill = purple](E,B)
\tkzDefTangent[from=B](F,A)
\tkzInterLL(F,tkzFirstPointResult)(C,D)
\tkzInterLL(A,tkzPointResult)(F,E)
\tkzDrawCircle[fill = yellow](tkzPointResult,Q)
\tkzDefPointBy[projection= onto B-A](tkzPointResult)
\tkzDrawCircle[fill = blue!50!black](tkzPointResult,A)
\end{tikzpicture}
```

9.3 绘制直线

只需要给出一条直线上的一对点，就可以绘制一条直线。如果直线由三角形定义，则参数是 3 个点的列表 (三角形的 3 个顶点)。其中，第 2 个点是直线的起点

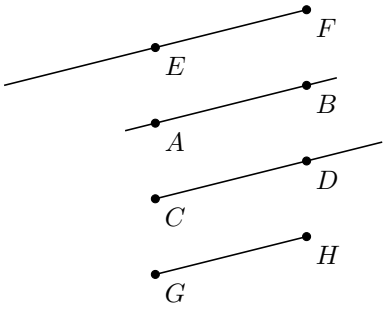
\tkzDrawLine[选项]($pt1, pt2$) or ($pt1, pt2, pt3$) 复数形式 \tkzDrawLines[选项]($pt1, pt2 \quad pt3, pt4 \quad \ldots$)

参数是 2 个或 3 个点

选项	默认值	含义
median	无	[median](A,B,C) 基点 B 处的中线
altitude	无	[altitude](C,A,B) 顶点 A 的高
bisector	无	[bisector](B,C,A) 顶点 C 的角平分线
none	无	绘制直线 (AB)
add= nb1 and nb2	.2 and .2	延伸线段

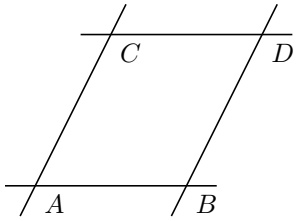
add 定义了通过点 pt1 和 pt2 的延伸长度，两个数字均为百分比

9.3.1 add 选项示例



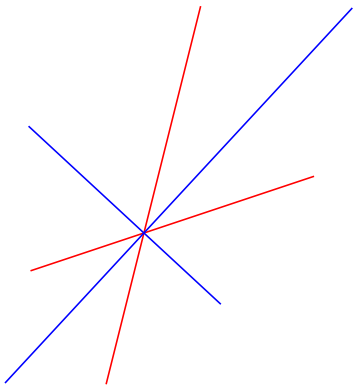

```
\begin{tikzpicture}
\tkzInit[xmin=-2,xmax=3,ymin=-2.25,ymax=2.25]
\tkzClip[space=.25]
\tkzDefPoint(0,0){A}  \tkzDefPoint(2,0.5){B}
\tkzDefPoint(0,-1){C} \tkzDefPoint(2,-0.5){D}
\tkzDefPoint(0,1){E}  \tkzDefPoint(2,1.5){F}
\tkzDefPoint(0,-2){G} \tkzDefPoint(2,-1.5){H}
\tkzDrawLine(A,B)
\tkzDrawLine[add = 0 and .5](C,D)
\tkzDrawLine[add = 1 and 0](E,F)
\tkzDrawLine[add = 0 and 0](G,H)
\tkzDrawPoints(A,B,C,D,E,F,G,H)
\tkzLabelPoints(A,B,C,D,E,F,G,H)
\end{tikzpicture}
```

9.3.2 tkzDrawLines 命令示例



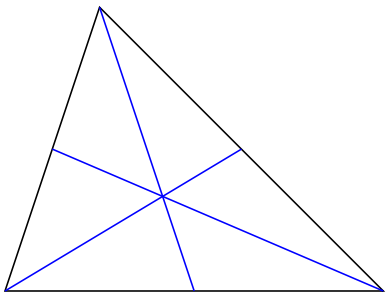

```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,0){B}
\tkzDefPoint(1,2){C}
\tkzDefPoint(3,2){D}
\tkzDrawLines(A,B C,D A,C B,D)
\tkzLabelPoints(A,B,C,D)
\end{tikzpicture}
```

9.3.3 add 选项示例



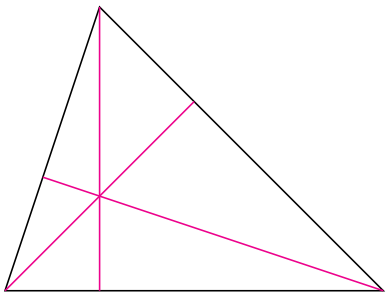
```
\begin{tikzpicture}[scale=.5]
\tkzDefPoints[]{0/0/O,3/1/I,1/4/J}
\tkzDefLine[bisector](I,O,J)
\tkzGetPoint{i}
\tkzDefLine[bisector out](I,O,J)
\tkzGetPoint{j}
\tkzDrawLines[add = 1 and .5,color=red](O,I O,J)
\tkzDrawLines[add = 1 and .5,color=blue](O,i O,j)
\end{tikzpicture}
```

9.3.4 三角形的中线示例



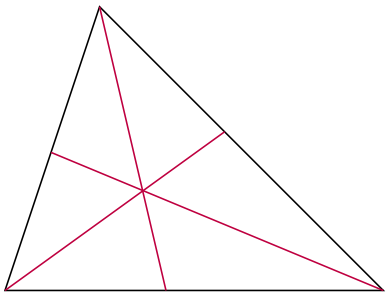
```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints[]{0/0/A,4/0/B,1/3/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpLine[color=blue]
\tkzDrawLines[median](B,C,A C,A,B A,B,C)
\end{tikzpicture}
```

9.3.5 三角形的高示例



```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints[]{0/0/A,4/0/B,1/3/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpLine[color=magenta]
\tkzDrawLines[altitude](B,C,A C,A,B A,B,C)
\end{tikzpicture}
```

9.3.6 三角形的角平分线示例



```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints[]{0/0/A,4/0/B,1/3/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpLine[color=purple]
\tkzDrawLines[bisector](B,C,A C,A,B A,B,C)
\end{tikzpicture}
```

9.4 标注直线

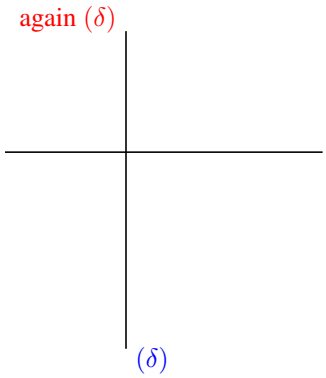
\tkzLabelLine[选项](*pt1*, *pt2*){*label*}

参数	默认值	含义
label		\tkzLabelLine(A,B){$ \Delta $}
选项	**默认值**	**含义**
pos	.5	pos 是 TikZ 的一个选项

可以使用 *pos* 外，所有有效 *TikZ* 样式，特别是用于设置标注位置的 *above*、*right*、…等样式选项

9.4.1 tkzLabelLine 命令示例

pos 是一个重要的选项，该选项的取值可以是大于 1，也可以是负值。




```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,3/0/B,1/1/C}
\tkzDefLine[perpendicular=through C,K=-1](A,B)
\tkzGetPoint{c}
\tkzDrawLines(A,B C,c)
\tkzLabelLine[pos=1.25,blue,right](C,c){$(\delta)$}
\tkzLabelLine[pos=-0.25,red,left](C,c){again $(\delta)$}
\end{tikzpicture}
```

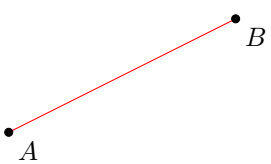
9.5 绘制线段

\tkzDrawSegment[选项](*pt1*, *pt2*)

参数	样例	含义
(*pt1*, *pt2*)	(A,B)	绘制线段 [A,B]
选项	**样例**	**含义**
TikZ 选项		所有有效 TikZ 选项
add	0 and 0	add = *kl* and *kr*, 允许线段向左右扩展
dim	无	dim = {label,dim,option}, 允许为图形添加尺寸

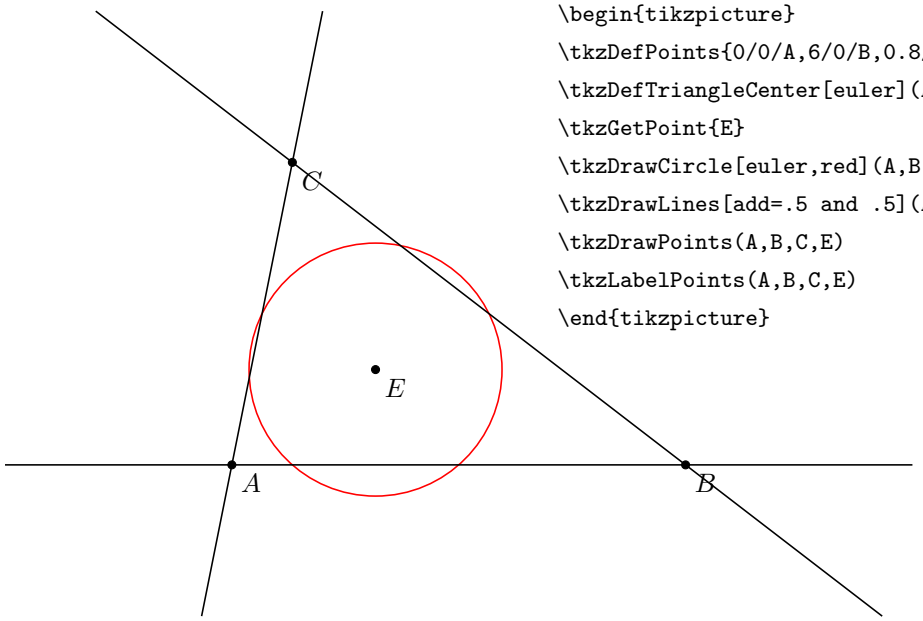
该命令等效于 *draw (A)--(B);* 命令

9.5.1 简单示例



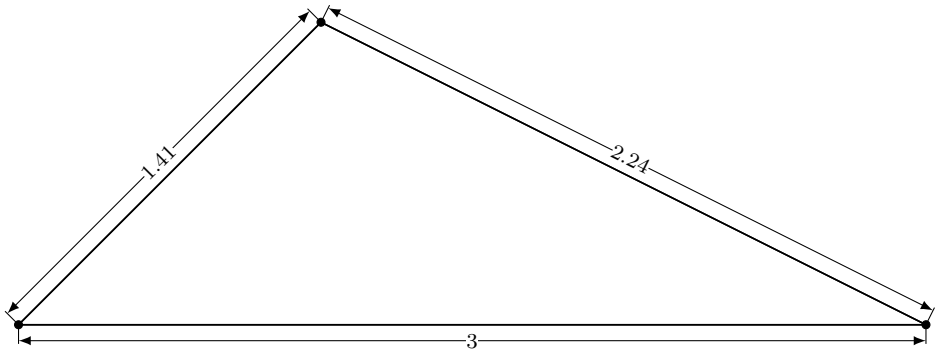

```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,1){B}
\tkzDrawSegment[color=red,thin](A,B)
\tkzDrawPoints(A,B)
\tkzLabelPoints(A,B)
\end{tikzpicture}
```

9.5.2 使用 add 选项延伸线段




```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefTriangleCenter[euler](A,B,C)
\tkzGetPoint{E}
\tkzDrawCircle[euler,red](A,B,C)
\tkzDrawLines[add=.5 and .5](A,B A,C B,C)
\tkzDrawPoints(A,B,C,E)
\tkzLabelPoints(A,B,C,E)
\end{tikzpicture}
```

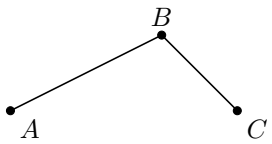
9.5.3 使用 dim 选项标注尺寸




```
\begin{tikzpicture}[scale=4]
\pgfkeys{/pgf/number format/.cd,fixed,precision=2}
% 定义已知点
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefPoint(1,1){C}
% 绘制三角形和点
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
% 计算长度
\tkzCalcLength[cm](A,B)\tkzGetLength{ABl}
\tkzCalcLength[cm](B,C)\tkzGetLength{BCl}
\tkzCalcLength[cm](A,C)\tkzGetLength{ACl}
% 添加尺寸标注
\tkzDrawSegment[dim={\pgfmathprintnumber\BCl,6pt,transform shape}](C,B)
\tkzDrawSegment[dim={\pgfmathprintnumber\ACl,6pt,transform shape}](A,C)
\tkzDrawSegment[dim={\pgfmathprintnumber\ABl,-6pt,transform shape}](A,B)
\end{tikzpicture}
```

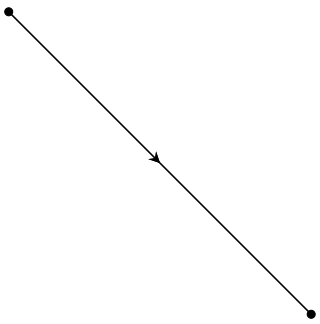
9.6 绘制多条线段

\tkzDrawSegments[选项]($pt1, pt2 \quad pt2, pt3 \ldots$)
参数是一组用空格分隔的线段的端点列表，两个端点之间用逗号分隔。在绘制中可以使用所有有效 TikZ 样式




```
\begin{tikzpicture}
\tkzInit[xmin=-1,xmax=3,ymin=-1,ymax=2]
\tkzClip[space=1]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,1){B}
\tkzDefPoint(3,0){C}
\tkzDrawSegments(A,B B,C)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,C)
\tkzLabelPoints[above](B)
\end{tikzpicture}
```

9.7 为线段添加箭头



```
\begin{tikzpicture}
\tikzset{
arr/.style={postaction=decorate,
decoration={markings,mark=at position .5 with {\arrow[thick]{#1}}}
}}
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,-4){B}
\tkzDrawSegments[arr=stealth](A,B)
\tkzDrawPoints(A,B)
\end{tikzpicture}
```

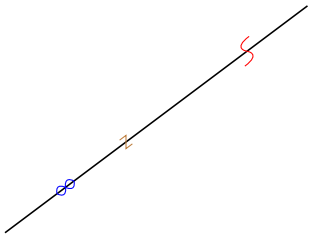
9.8 标记线段

该命令用于为线段添加标记

\tkzMarkSegment[选项]($pt1, pt2$) 复数形式 \tkzMarkSegments[选项]($pt1, pt2 \quad pt3, pt4 \quad \ldots$)

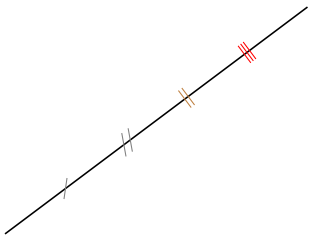
选项	默认值	含义
pos	.5	标记位置
color	black	标记颜色
mark	none	标记类型
size	4pt	标记尺寸

9.8.1 几种标记示例



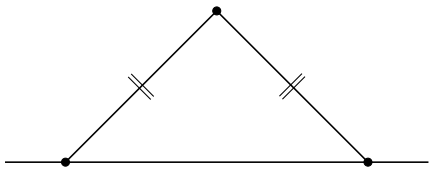
```
\begin{tikzpicture}
\tkzDefPoint(2,1){A}
\tkzDefPoint(6,4){B}
\tkzDrawSegment(A,B)
\tkzMarkSegment[color=brown,size=2pt,pos=0.4, mark=z](A,B)
\tkzMarkSegment[color=blue,pos=0.2, mark=oo](A,B)
\tkzMarkSegment[pos=0.8,mark=s,color=red](A,B)
\end{tikzpicture}
```

9.8.2 mark 选项示例



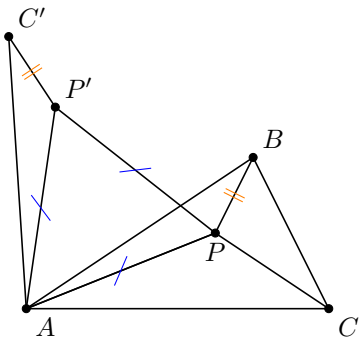
```
\begin{tikzpicture}
\tkzDefPoint(2,1){A}
\tkzDefPoint(6,4){B}
\tkzDrawSegment(A,B)
\tkzMarkSegment[color=gray,pos=0.2,mark=s|](A,B)
\tkzMarkSegment[color=gray,pos=0.4,mark=s||](A,B)
\tkzMarkSegment[color=brown,pos=0.6,mark=||](A,B)
\tkzMarkSegment[color=red,pos=0.8,mark=|||](A,B)
\end{tikzpicture}
```

9.8.3 标记等腰三角形示例



```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/O,2/2/A,4/0/B,6/2/C}
\tkzDrawSegments(O,A A,B)
\tkzDrawPoints(O,A,B)
\tkzDrawLine(O,B)
\tkzMarkSegments[mark=||,size=6pt](O,A A,B)
\end{tikzpicture}
```

9.8.4 其它标记示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,0){A}\tkzDefPoint(3,2){B}
\tkzDefPoint(4,0){C}\tkzDefPoint(2.5,1){P}
\tkzDrawPolygon(A,B,C)
\tkzDefEquilateral(A,P) \tkzGetPoint{P'}
\tkzDefPointsBy[rotation=center A angle 60](P,B){P',C'}
\tkzDrawPolygon(A,P,P')
\tkzDrawPolySeg(P',C',A,P,B)
\tkzDrawSegment(C,P)
\tkzDrawPoints(A,B,C,C',P,P')
\tkzMarkSegments[mark=s|,size=6pt,color=blue](A,P P,P' P',A)
\tkzMarkSegments[mark=||,color=orange](B,P P',C')
\tkzLabelPoints(A,C) \tkzLabelPoints[below](P)
\tkzLabelPoints[above right](P',C',B)
\end{tikzpicture}
```

9.9 标注线段

\tkzLabelSegment[选项](*pt*1, *pt*2){label} 复数形式 \tkzLabelSegments[选项](*pt*1, *pt*2　*pt*3, *pt*4　...)

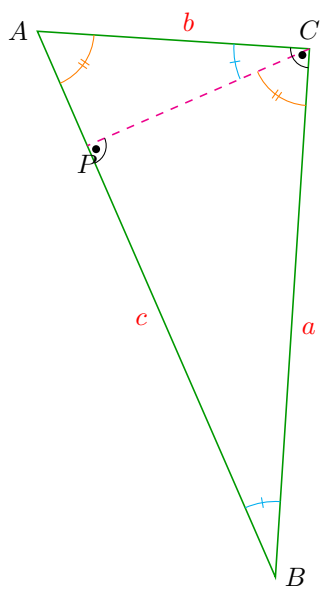
参数	样例	含义
(*pt*1, *pt*2)	(A,B)	被标注线段 $[AB]$
label	\tkzLabelSegment(A,B){5}	标注文本
选项	**样例**	**含义**
pos	.5	标注的位置

9.9.1 多个标注示例

$$\frac{a}{4}$$

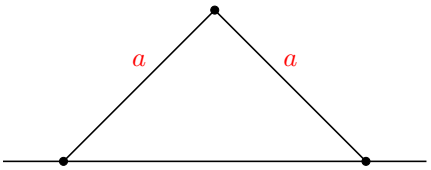
```
\begin{tikzpicture}
\tkzInit
\tkzDefPoint(0,0){A}
\tkzDefPoint(6,0){B}
\tkzDrawSegment(A,B)
\tkzLabelSegment[above,pos=.8](A,B){$a$}
\tkzLabelSegment[below,pos=.2](A,B){$4$}
\end{tikzpicture}
```

9.9.2 直角三角形的标注和标记示例




```
\begin{tikzpicture}[rotate=-60]
\tikzset{label seg style/.append style = {color=red,}}
\tkzDefPoint(0,1){A}
\tkzDefPoint(2,4){C}
\tkzDefPointWith[orthogonal normed,K=7](C,A)
\tkzGetPoint{B}
\tkzDrawPolygon[green!60!black](A,B,C)
\tkzDrawLine[altitude,dashed,color=magenta](B,C,A)
\tkzGetPoint{P}
\tkzLabelPoint[left](A){$A$}
\tkzLabelPoint[right](B){$B$}
\tkzLabelPoint[above](C){$C$}
\tkzLabelPoint[below](P){$P$}
\tkzLabelSegment[](B,A){$c$}
\tkzLabelSegment[swap](B,C){$a$}
\tkzLabelSegment[swap](C,A){$b$}
\tkzMarkAngles[size=1cm,color=cyan,mark=|](C,B,A A,C,P)
\tkzMarkAngle[size=0.75cm,color=orange,mark=||](P,C,B)
\tkzMarkAngle[size=0.75cm,color=orange,mark=||](B,A,C)
\tkzMarkRightAngles[german](A,C,B B,P,C)
\end{tikzpicture}
```

9.9.3 等腰三角形的标注示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/O,2/2/A,4/0/B,6/2/C}
\tkzDrawSegments(O,A A,B)
\tkzDrawPoints(O,A,B)
\tkzDrawLine(O,B)
\tkzLabelSegments[color=red,above=4pt](O,A A,B){$a$}
\end{tikzpicture}
```

第十章 定义与绘制三角形

10.1 定义三角形

三角形定义命令允许使用**至少** 2 个点构造一个三角形。可以按如下方式定义三角形：

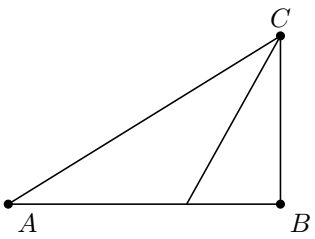
- two angles 已知 2 个角的三角形;
- equilateral 等边三角形;
- half 直角边之和与斜边之比等于 2 的直角三角形;
- pythagore 勾股直角三角形;
- school 三个角分别是 30、60 和 90 的直角三角形;
- golden 直角边之和与斜边比等于黄金分割比 $\Phi = 1.618034$ 的直角三角形;
- euclide or gold 底角为 72 度的等腰三角形称为 “golden triangle” 或 “Euclid’s triangle”;
- cheops 三边为 2、Φ 和 Φ 的等腰三角形.

\tkzDefTriangle[选项](A,B)

参数中的点的顺序决定的另外一个点的位置

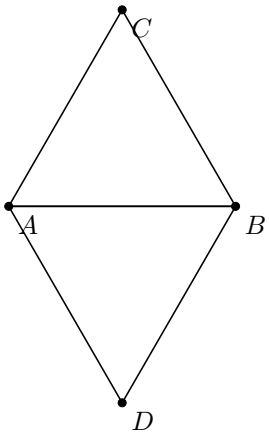
选项	默认值	含义
two angles= #1 and #2	无	三角形两个已知角
equilateral	无	等边三角形
pythagore	无	勾股三角形
school	无	三个角分别是 30、60 和 90 度
gold	无	三个角分别是 72、72 和 36 度，A 是顶点
euclide	无	同上，但 $[AB]$ 是底边
golden	无	AB 构成矩形，并且 $AB/BC = \Phi$
cheops	无	$AC = BC$，AC 和 BC 及第三边满足 2 和 Φ 的比例关系

10.1.1 golden 选项示例



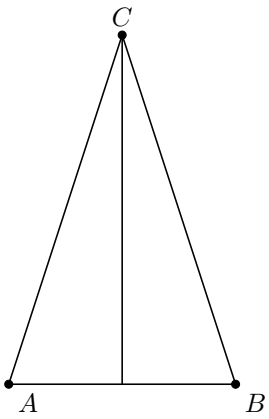

```
\begin{tikzpicture}[scale=0.9]
\tkzInit[xmax=5,ymax=3]
\tkzClip[space=.5]
\tkzDefPoint(0,0){A} \tkzDefPoint(4,0){B}
\tkzDefTriangle[golden](A,B)\tkzGetPoint{C}
\tkzDrawPolygon(A,B,C) \tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,B) \tkzDrawBisector(A,C,B)
\tkzLabelPoints[above](C)
\end{tikzpicture}
```

10.1.2 equilateral 选项示例




```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefTriangle[equilateral](A,B)
\tkzGetPoint{C}
\tkzDrawPolygon(A,B,C)
\tkzDefTriangle[equilateral](B,A)
\tkzGetPoint{D}
\tkzDrawPolygon(B,A,D)
\tkzDrawPoints(A,B,C,D)
\tkzLabelPoints(A,B,C,D)
\end{tikzpicture}
```

10.1.3 gold 或 euclide 选项示例




```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDefTriangle[euclide](A,B)
\tkzGetPoint{C}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,B)
\tkzLabelPoints[above](C)
\tkzDrawBisector(A,C,B)
\end{tikzpicture}
```

10.2 绘制三角形

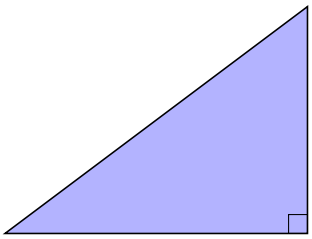
\tkzDrawTriangle[选项](A,B)

与三角形定义命令类似，但可绘制三角形

选项	默认值	含义
two angles= #1 and #2	无	三角形两个已知角
equilateral	无	等边三角形
pythagore	无	勾股三角形
school	无	三个角分别是 30、60 和 90 度
gold	无	三个角分别是 72、72 和 36 度，A 是顶点
euclide	无	同上，但 $[AB]$ 是底边
golden	无	AB 构成矩形，并且 $AB/BC = \Phi$
cheops	无	$AC = BC$，AC 和 BC 及第三边满足 2 和 Φ 的比例关系

10.2.1 pythagore 选项示例

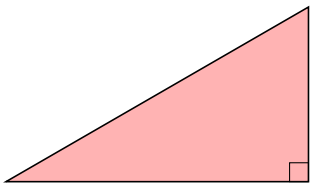
三角形的三个边满足类似 3、4 和 5 的勾股关系。



```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDrawTriangle[pythagore,fill=blue!30](A,B)
\tkzMarkRightAngles(A,B,tkzPointResult)
\end{tikzpicture}
```

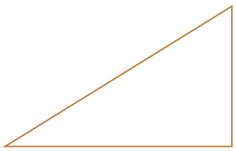
10.2.2 school 选项示例

三角形的三个内角分别是 30、60 和 90 度。



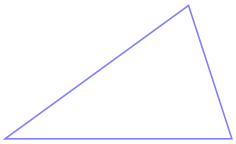
```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDrawTriangle[school,fill=red!30](A,B)
\tkzMarkRightAngles(tkzPointResult,B,A)
\end{tikzpicture}
```

10.2.3 golden 选项示例



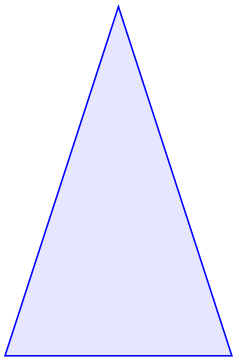
```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,-10){M}
\tkzDefPoint(3,-10){N}
\tkzDrawTriangle[golden,color=brown](M,N)
\end{tikzpicture}
```

10.2.4 gold 选项示例



```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(5,-5){I}
\tkzDefPoint(8,-5){J}
\tkzDrawTriangle[gold,color=blue!50](I,J)
\end{tikzpicture}
```

10.2.5 euclide 选项示例



```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(10,-5){K}
\tkzDefPoint(13,-5){L}
\tkzDrawTriangle[euclide,color=blue,fill=blue!10](K,L)
\end{tikzpicture}
```

10.3 定义特殊三角形

在“点”的定义小节中，定义了一些三角形中的特殊点，在此，可以使用这些点确定三角形

\tkzDefSpcTriangle[选项](A, B, C)

注意，点的顺序决定了计算结果

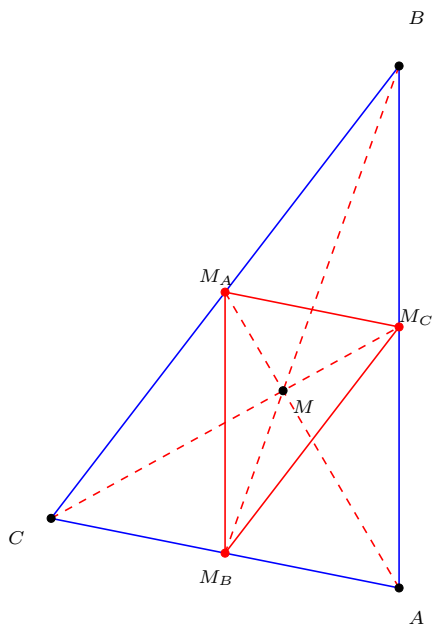
选项	默认值	含义
in or incentral	centroid	内心三角形
ex or excentral	centroid	旁心三角形
extouch	centroid	外切三角形
intouch or contact	centroid	内切三角形
centroid or medial	centroid	三边中点构成的三角形
orthic	centroid	正交三角形
feuerbach	centroid	九点圆与旁切圆的切点构成的三角形
euler	centroid	欧拉三角形
tangential	centroid	切身三角形
name	无	顶点命名前缀

10.3.1 centroid 或 medial 选项示例

三角形的质心用 G 表示 (有时也用 M 表示)，它是三角形三条中线的交点，该点也称为重心，重心总是位于三角形内部。

Weisstein, Eric W. “Centroid triangle” From MathWorld–A Wolfram Web Resource.

下面的例子中，通过预先定义的点，得到通过这些点的欧拉圆。

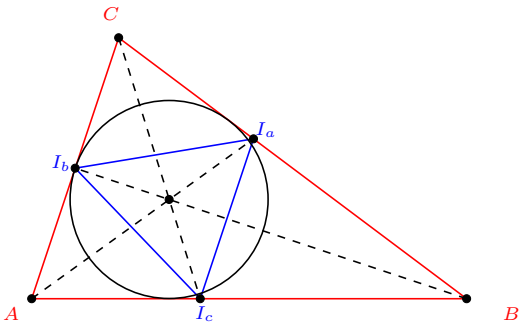



```
\begin{tikzpicture}[rotate=90,scale=1.15]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefTriangleCenter[centroid](A,B,C)
\tkzGetPoint{M}
\tkzDefSpcTriangle[medial,name=M](A,B,C){_A,_B,_C}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawSegments[dashed,red](A,M_A B,M_B C,M_C)
\tkzDrawPolygon[color=red](M_A,M_B,M_C)
\tkzDrawPoints(A,B,C,M)
\tkzDrawPoints[red](M_A,M_B,M_C)
\tkzAutoLabelPoints[center=M,font=\scriptsize]%
(A,B,C,M_A,M_B,M_C)
\tkzLabelPoints[font=\scriptsize](M)
\end{tikzpicture}
```

10.3.2 in 或 incentral 选项示例

内心三角形是由一个三角形的三个内角平分线与对边交点确定的三角形。

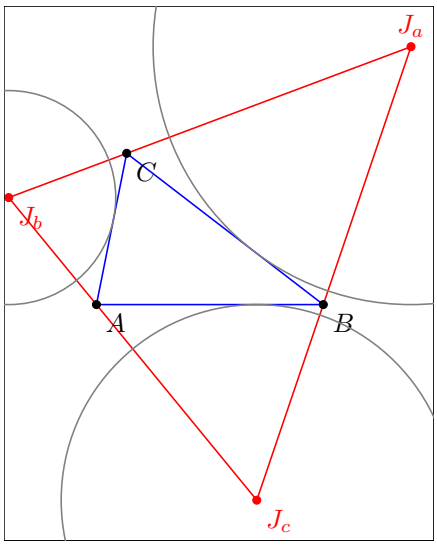
Weisstein, Eric W. “Incentral triangle” From MathWorld–A Wolfram Web Resource.




```
\begin{tikzpicture}[scale=1.15]
\tkzDefPoints{ 0/0/A,5/0/B,1/3/C}
\tkzDefSpcTriangle[in,name=I](A,B,C){_a,_b,_c}
\tkzInCenter(A,B,C)
\tkzGetPoint{I}
\tkzDrawPolygon[red](A,B,C)
\tkzDrawPolygon[blue](I_a,I_b,I_c)
\tkzDrawPoints(A,B,C,I,I_a,I_b,I_c)
\tkzDrawCircle[in](A,B,C)
\tkzDrawSegments[dashed](A,I_a B,I_b C,I_c)
\tkzAutoLabelPoints[center=I,blue,font=\scriptsize](I_a,I_b,I_c)
\tkzAutoLabelPoints[center=I,red,font=\scriptsize](A,B,C)
\end{tikzpicture}
```

10.3.3 ex 或 excentral 选项示例

旁心三角形是由一个三角形的三个旁心构成的三角形。

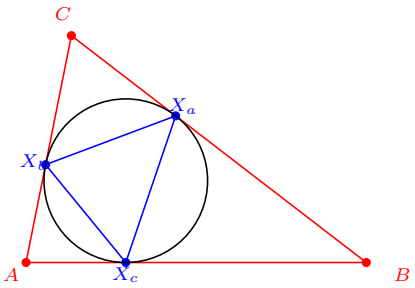



```
\begin{tikzpicture}[scale=0.50]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefSpcTriangle[excentral,name=J](A,B,C){_a,_b,_c}
\tkzDefSpcTriangle[extouch,name=T](A,B,C){_a,_b,_c}
\tkzDrawPolygon[blue](A,B,C)
\tkzDrawPolygon[red](J_a,J_b,J_c)
\tkzDrawPoints(A,B,C)
\tkzDrawPoints[red](J_a,J_b,J_c)
\tkzLabelPoints(A,B,C)
\tkzLabelPoints[red](J_b,J_c)
\tkzLabelPoints[red,above](J_a)
\tkzClipBB
\tkzShowBB
\tkzDrawCircles[gray](J_a,T_a J_b,T_b J_c,T_c)
\end{tikzpicture}
```

10.3.4 intouch 选项示例

内接三角形是由三角形的内切圆的三个切点构成的三角形。

Weisstein, Eric W. “Contact triangle” From MathWorld–A Wolfram Web Resource.



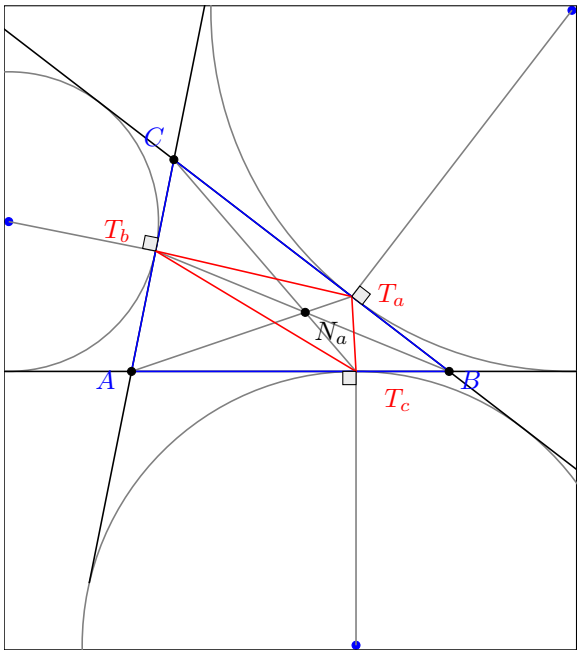

```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefSpcTriangle[intouch,name=X](A,B,C){_a,_b,_c}
\tkzInCenter(A,B,C)\tkzGetPoint{I}
\tkzDrawPolygon[red](A,B,C)
\tkzDrawPolygon[blue](X_a,X_b,X_c)
\tkzDrawPoints[red](A,B,C)
\tkzDrawPoints[blue](X_a,X_b,X_c)
\tkzDrawCircle[in](A,B,C)
\tkzAutoLabelPoints[center=I,blue,font=\scriptsize](X_a,X_b,X_c)
\tkzAutoLabelPoints[center=I,red,font=\scriptsize](A,B,C)
\end{tikzpicture}
```

10.3.5 extouch 选项示例

外切三角形 $T_aT_bT_c$ 是由三角形 ABC 的三个旁切圆 J_a、J_b 和 J_c 的切点构成的三角形。

Weisstein, Eric W. “Extouch triangle” From MathWorld–A Wolfram Web Resource.

可以得到旁切圆的切点和由这三个切点构成的三角形。

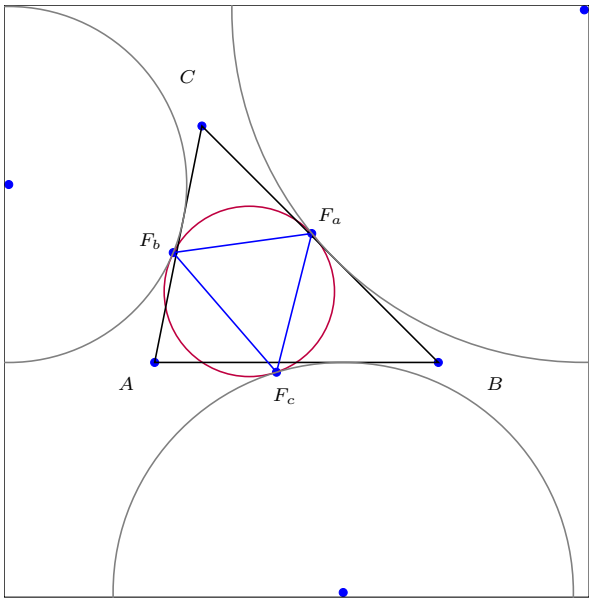



```
\begin{tikzpicture}[scale=.7]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefSpcTriangle[excentral,
name=J](A,B,C){_a,_b,_c}
\tkzDefSpcTriangle[extouch,
name=T](A,B,C){_a,_b,_c}
\tkzDefTriangleCenter[nagel](A,B,C)
\tkzGetPoint{N_a}
\tkzDefTriangleCenter[centroid](A,B,C)
\tkzGetPoint{G}
\tkzDrawPoints[blue](J_a,J_b,J_c)
\tkzClipBB \tkzShowBB
\tkzDrawCircles[gray](J_a,T_a J_b,T_b J_c,T_c)
\tkzDrawLines[add=1 and 1](A,B B,C C,A)
\tkzDrawSegments[gray](A,T_a B,T_b C,T_c)
\tkzDrawSegments[gray](J_a,T_a J_b,T_b J_c,T_c)
\tkzDrawPolygon[blue](A,B,C)
\tkzDrawPolygon[red](T_a,T_b,T_c)
\tkzDrawPoints(A,B,C,N_a)
\tkzLabelPoints(N_a)
\tkzAutoLabelPoints[center=N_a,blue](A,B,C)
\tkzAutoLabelPoints[center=G,red,
dist=.4](T_a,T_b,T_c)
\tkzMarkRightAngles[fill=gray!15](J_a,T_a,B
J_b,T_b,C J_c,T_c,A)
\end{tikzpicture}
```

10.3.6 feuerbach 选项示例

Feuerbach 三角形是由九点圆与三个旁切圆的 3 个切点构成的三角形。

Weisstein, Eric W. “Feuerbach triangle” From MathWorld–A Wolfram Web Resource.

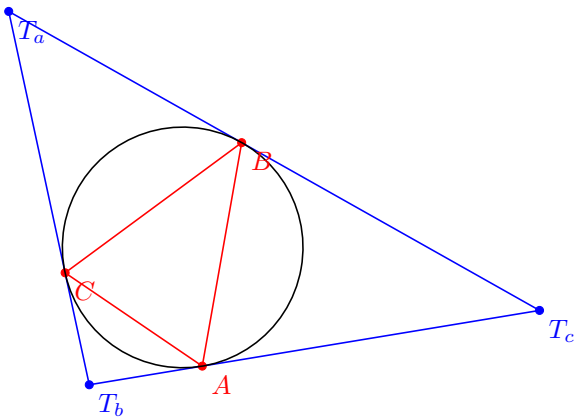



```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefPoint(0.5,2.5){C}
\tkzDefCircle[euler](A,B,C) \tkzGetPoint{N}
\tkzDefSpcTriangle[feuerbach,
name=F](A,B,C){_a,_b,_c}
\tkzDefSpcTriangle[excentral,
name=J](A,B,C){_a,_b,_c}
\tkzDefSpcTriangle[extouch,
name=T](A,B,C){_a,_b,_c}
\tkzDrawPoints[blue](J_a,J_b,J_c,F_a,F_b,F_c,A,B,C)
\tkzClipBB
\tkzShowBB
\tkzDrawCircle[purple](N,F_a)
\tkzDrawPolygon(A,B,C)
\tkzDrawPolygon[blue](F_a,F_b,F_c)
\tkzDrawCircles[gray](J_a,F_a J_b,F_b J_c,F_c)
\tkzAutoLabelPoints[center=N,dist=.3,
font=\scriptsize](A,B,C,F_a,F_b,F_c,J_a,J_b,J_c)
\end{tikzpicture}
```

10.3.7 tangential 选项示例

切向三角形是三角形 ABC 外接圆在三个顶点处的切线构成的三角形 $T_aT_bT_c$。因此，它是相对于三角形 ABC 外心的反三角三角形。

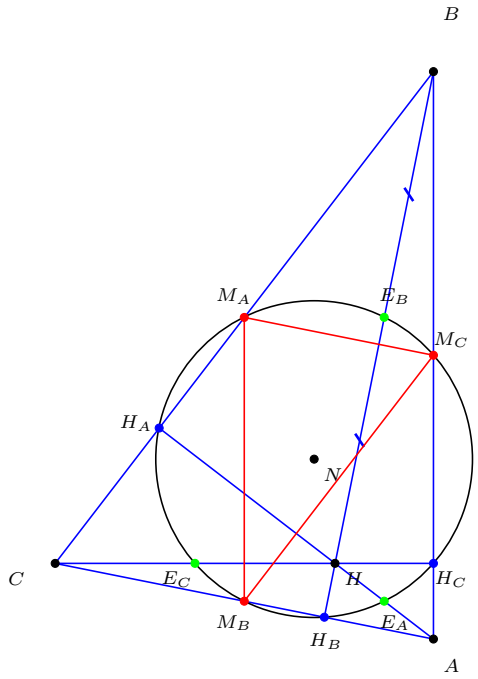
Weisstein, Eric W. “Tangential Triangle” From MathWorld–A Wolfram Web Resource.




```
\begin{tikzpicture}[scale=.5,rotate=80]
\tkzDefPoints{0/0/A,6/0/B,1.8/4/C}
\tkzDefSpcTriangle[tangential,
name=T](A,B,C){_a,_b,_c}
\tkzDrawPolygon[red](A,B,C)
\tkzDrawPolygon[blue](T_a,T_b,T_c)
\tkzDrawPoints[red](A,B,C)
\tkzDrawPoints[blue](T_a,T_b,T_c)
\tkzDefCircle[circum](A,B,C)
\tkzGetPoint{O}
\tkzDrawCircle(O,A)
\tkzLabelPoints[red](A,B,C)
\tkzLabelPoints[blue](T_a,T_b,T_c)
\end{tikzpicture}
```

10.3.8 euler 选项示例

欧拉三角形是由三角形 ABC 的垂心 H 与三个顶点连线中点构成的三角形 $E_AE_BE_C$，欧拉三角形的顶点是欧拉点，它们位于三角形的九点圆上。




```
\begin{tikzpicture}[rotate=90,scale=1.25]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefSpcTriangle[medial,
name=M](A,B,C){_A,_B,_C}
\tkzDefTriangleCenter[euler](A,B,C)
\tkzGetPoint{N} % I= N nine points
\tkzDefTriangleCenter[ortho](A,B,C)
\tkzGetPoint{H}
\tkzDefMidPoint(A,H) \tkzGetPoint{E_A}
\tkzDefMidPoint(C,H) \tkzGetPoint{E_C}
\tkzDefMidPoint(B,H) \tkzGetPoint{E_B}
\tkzDefSpcTriangle[ortho,name=H](A,B,C){_A,_B,_C}
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawCircle(N,E_A)
\tkzDrawSegments[blue](A,H_A B,H_B C,H_C)
\tkzDrawPoints(A,B,C,N,H)
\tkzDrawPoints[red](M_A,M_B,M_C)
\tkzDrawPoints[blue]( H_A,H_B,H_C)
\tkzDrawPoints[green](E_A,E_B,E_C)
\tkzAutoLabelPoints[center=N,font=\scriptsize]%
(A,B,C,M_A,M_B,M_C,H_A,H_B,H_C,E_A,E_B,E_C)
\tkzLabelPoints[font=\scriptsize](H,N)
\tkzMarkSegments[mark=s|,size=3pt,
color=blue,line width=1pt](B,E_B E_B,H)
\tkzDrawPolygon[color=red](M_A,M_B,M_C)
\end{tikzpicture}
```

第十一章 定义与绘制多边形

11.1 定义正方形

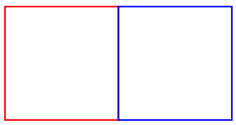
\tkzDefSquare($pt1, pt2$)

通过两个点按逆时针方向推算另外两个点后，得到正方形

参数	样例	说明
$pt1, pt2$	\tkzDefSquare(A, B)	按指定的方向定义正方形

11.1.1 通过两个点定义正方形

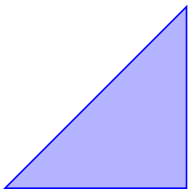
需要注意点的方向问题。



```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,0){A} \tkzDefPoint(0,3){B}
\tkzDefSquare(A,B)
\tkzDrawPolygon[color=red](A,B,tkzFirstPointResult,tkzSecondPointResult)
\tkzDefSquare(B,A)
\tkzDrawPolygon[color=blue](B,A,tkzFirstPointResult,tkzSecondPointResult)
\end{tikzpicture}
```

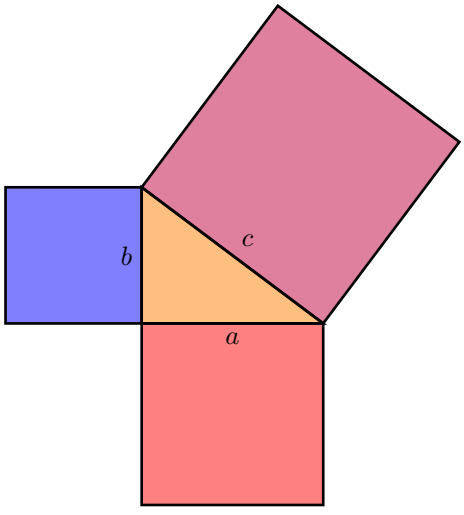
11.1.2 绘制等腰直角三角形

可以使用 tkzGetFirstPoint 或 tkzGetSecondPoint 命令利用其中的 1 个点绘制等腰直角三角形。



```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefSquare(A,B) \tkzGetFirstPoint{C}
\tkzDrawPolygon[color=blue,fill=blue!30](A,B,C)
\end{tikzpicture}
```

11.1.3 绘制 Pythagorean 定理示意图




```
\begin{tikzpicture}[scale=.6]%\tkzInit
\tkzDefPoints{0/0/C,4/0/A,0/3/B}
\tkzDefSquare(B,A)\tkzGetPoints{E}{F}
\tkzDefSquare(A,C)\tkzGetPoints{G}{H}
\tkzDefSquare(C,B)\tkzGetPoints{I}{J}
\tkzFillPolygon[fill = red!50 ](A,C,G,H)
\tkzFillPolygon[fill = blue!50 ](C,B,I,J)
\tkzFillPolygon[fill = purple!50](B,A,E,F)
\tkzFillPolygon[fill = orange,opacity=.5](A,B,C)
\tkzDrawPolygon[line width = 1pt](A,B,C)
\tkzDrawPolygon[line width = 1pt](A,C,G,H)
\tkzDrawPolygon[line width = 1pt](C,B,I,J)
\tkzDrawPolygon[line width = 1pt](B,A,E,F)
\tkzLabelSegment(A,C){$a$}\tkzLabelSegment(C,B){$b$}
\tkzLabelSegment[swap](A,B){$c$}
\end{tikzpicture}
```

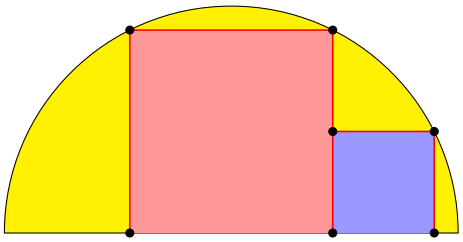
11.2 绘制正方形

\tkzDrawSquare[选项]($pt1, pt2$)

用于绘制一个正方形，但不绘制顶点。可以对内部进行着色，点的顺序是逆时针方向

参数	样例	说明
$pt1, pt2$	\tkzDrawSquare(A, B)	\tkzGetPointsCD
选项	**样例**	**说明**
TikZ 选项	red,line width=1pt	所有有效 TikZ 选项

11.2.1 在半圆内绘制两个正方形示例



```
\begin{tikzpicture}[scale=.75]
\tkzInit[ymax=8,xmax=8]
\tkzClip[space=.25]   \tkzDefPoint(0,0){A}
\tkzDefPoint(8,0){B}  \tkzDefPoint(4,0){I}
\tkzDefSquare(A,B)    \tkzGetPoints{C}{D}
\tkzInterLC(I,C)(I,B) \tkzGetPoints{E'}{E}
\tkzInterLC(I,D)(I,B) \tkzGetPoints{F'}{F}
\tkzDefPointsBy[projection=onto A-B](E,F){H,G}
\tkzDefPointsBy[symmetry  = center H](I){J}
\tkzDefSquare(H,J)    \tkzGetPoints{K}{L}
\tkzDrawSector[fill=yellow](I,B)(A)
\tkzFillPolygon[color=red!40](H,E,F,G)
\tkzFillPolygon[color=blue!40](H,J,K,L)
\tkzDrawPolySeg[color=red](H,E,F,G)
\tkzDrawPolySeg[color=red](J,K,L)
\tkzDrawPoints(E,G,H,F,J,K,L)
\end{tikzpicture}
```

11.3 定义平行四边形

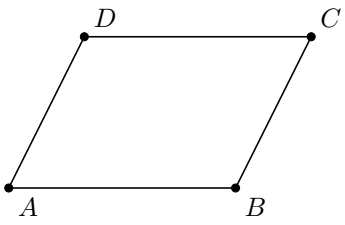
可以通过 3 个点定义一个平行四边形

\tkzDefParallelogram($pt1, pt2, pt3$)

通过 3 个点，计算另一个点，构成平行四边形

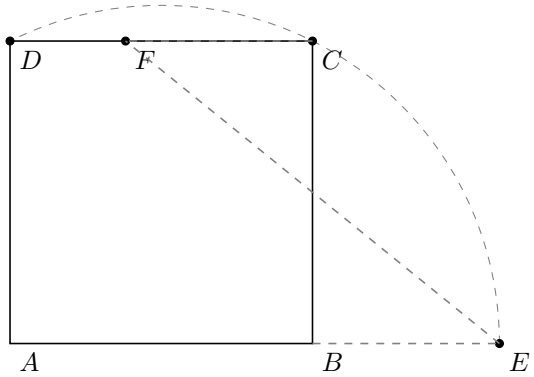
参数	默认值	含义
$pt1, pt2, pt3$	无	必须的 3 个顶点

11.3.1 平行四边形定义示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{0/0/A,3/0/B,4/2/C}
\tkzDefParallelogram(A,B,C)
\tkzGetPoint{D}
\tkzDrawPolygon(A,B,C,D)
\tkzLabelPoints(A,B)
\tkzLabelPoints[above right](C,D)
\tkzDrawPoints(A,...,D)
\end{tikzpicture}
```

11.3.2 黄金矩形示例




```
\begin{tikzpicture}[scale=.5]
\tkzInit[xmax=14,ymax=10]
\tkzClip[space=1]
\tkzDefPoint(0,0){A}
\tkzDefPoint(8,0){B}
\tkzDefMidPoint(A,B)\tkzGetPoint{I}
\tkzDefSquare(A,B)\tkzGetPoints{C}{D}
\tkzDrawSquare(A,B)
\tkzInterLC(A,B)(I,C)\tkzGetPoints{G}{E}
\tkzDrawArc[style=dashed,color=gray](I,E)(D)
\tkzDefPointWith[colinear= at C](E,B)
\tkzGetPoint{F}
\tkzDrawPoints(C,D,E,F)
\tkzLabelPoints(A,B,C,D,E,F)
\tkzDrawSegments[style=dashed,color=gray](E,F C,F B,E)
\end{tikzpicture}
```

11.4 定义黄金矩形

定义长宽比为黄金分割比 Φ 的黄金矩形

\tkzDefGoldRectangle($pt1, pt2$)

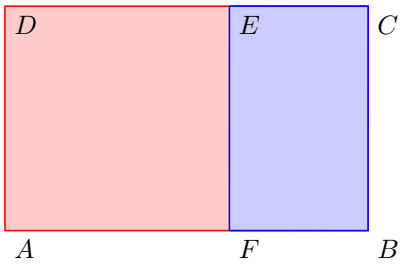
参数	样例	说明
$pt1, pt2$	\tkzDefGoldRectangle($pt1, pt2$)	如果用 C 和 D 表示得到的点，则 $AB/BC = \Phi$

11.5 绘制黄金矩形

\tkzDrawGoldRectangle[选项]($pt1, pt2$)

参数	样例	说明
$pt1, pt2$	\tkzDrawGoldRectangle($pt1, pt2$)	根据线段 $[AB]$ 绘制黄金矩形
选项	样例	说明
TikZ 选项	red,line width=1pt	所有有效 TikZ 选项

11.5.1 黄金矩形示例




```
\begin{tikzpicture}[scale=.6]
\tkzDefPoint(0,0){A}       \tkzDefPoint(8,0){B}
\tkzDefGoldRectangle(A,B) \tkzGetPoints{C}{D}
\tkzDefGoldRectangle(B,C) \tkzGetPoints{E}{F}
\tkzDrawPolygon[color=red,fill=red!20](A,B,C,D)
\tkzDrawPolygon[color=blue,fill=blue!20](B,C,E,F)
\tkzLabelPoints(A,B,C,D,E,F)
\end{tikzpicture}
```

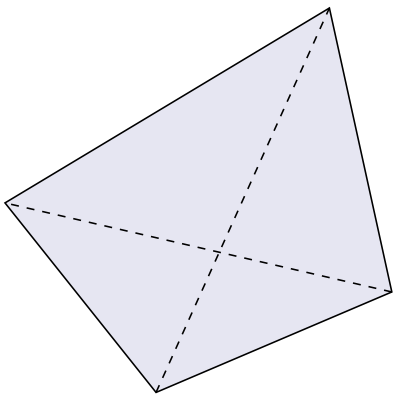
11.6 绘制多边形

\tkzDrawPolygon[选项](点集列表)

用给定的点集绘制多边形。连续的点可以省略中间的点，例: 可以使用 $(A,\ldots,E)$ 表示点集 (A,B,C,D,E)

参数	样例	说明
$(pt1,pt2,pt3,\ldots)$	\tkzDrawPolygon[gray,dashed](A,B,C)	绘制一个三角形
选项	**默认值**	**样例**
TikZ 选项	无	\tkzDrawPolygon[red,line width=2pt](A,B,C)

11.6.1 tkzDrawPolygon 命令示例



```
\begin{tikzpicture}[rotate=18,scale=1.5]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2.25,0.2){B}
\tkzDefPoint(2.5,2.75){C}
\tkzDefPoint(-0.75,2){D}
\tkzDrawPolygon[fill=black!50!blue!10](A,B,C,D)
\tkzDrawSegments[style=dashed](A,C B,D)
\end{tikzpicture}
```

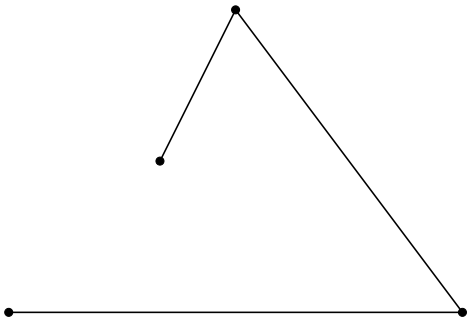
11.7 绘制多边形顶点折线

\tkzDrawPolySeg[选项](点集列表)

绘制多边形顶点构成的折线

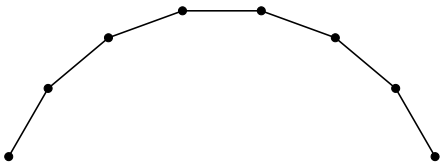
参数	样例	说明
$(pt1,pt2,pt3,\ldots)$	\tkzDrawPolySeg[gray,dashed](A,B,C)	绘制多顶点折线
选项	**默认值**	**样例**
TikZ 选项	无	\tkzDrawPolygon[red,line width=2pt](A,B,C)

11.7.1 多边形顶点折线示例



```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,6/0/B,3/4/C,2/2/D}
\tkzDrawPolySeg(A,...,D)
\tkzDrawPoints(A,...,D)
\end{tikzpicture}
```

11.7.2 多边形顶点折线：循环实现



```
\begin{tikzpicture}
\foreach \pt in {1,2,...,8} {%
\tkzDefPoint(\pt*20:3){P_\pt}}
\tkzDrawPolySeg(P_1,P_...,P_8)
\tkzDrawPoints(P_1,P_...,P_8)
\end{tikzpicture}
```

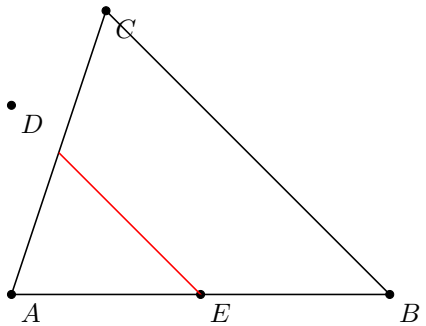
11.8 使用多边形裁剪

\tkzClipPolygon[选项](点集列表)

用指定的多边形对图形进行裁剪

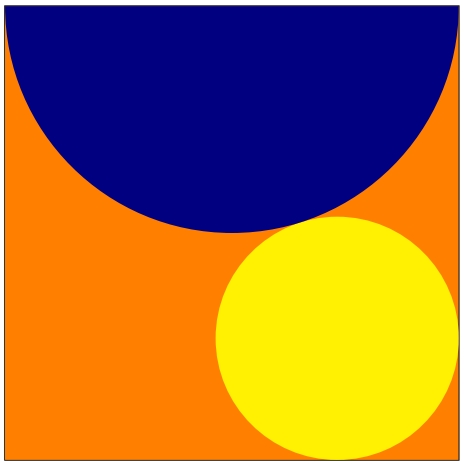
参数	样例	说明
$(pt1, pt2, pt3, \dots)$	\tkzClipPolygon(A,B,C)	用三角形 ABC 对图形进行剪裁

11.8.1 tkzClipPolygon 命令示例




```
\begin{tikzpicture}[scale=1.25]
\tkzInit[xmin=0,xmax=4,ymin=0,ymax=3]
\tkzClip[space=.5]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDefPoint(1,3){C}
\tkzDrawPolygon(A,B,C)
\tkzDefPoint(0,2){D}
\tkzDefPoint(2,0){E}
\tkzDrawPoints(A,...,E)
\tkzLabelPoints(A,...,E)
\tkzClipPolygon(A,B,C)
\tkzDrawLine[color=red](D,E)
\end{tikzpicture}
```

11.8.2 使用“裁剪”将 Sangaku 图形限制在正方形内



```
\begin{tikzpicture}[scale=.75]
\tkzDefPoint(0,0){A} \tkzDefPoint(8,0){B}
\tkzDefSquare(A,B) \tkzGetPoints{C}{D}
\tkzDrawPolygon(B,C,D,A)
\tkzClipPolygon(B,C,D,A)
\tkzDefMidPoint(C,D)
\tkzGetPoint{F}
\tkzDefTriangle[equilateral](C,D)
\tkzGetPoint{I}
\tkzDefPointBy[projection=onto B-C](I)
\tkzGetPoint{J}
\tkzInterLL(D,B)(I,J)  \tkzGetPoint{K}
\tkzDefPointBy[symmetry=center K](B)
\tkzGetPoint{M}
\tkzDrawCircle(M,I)
\tkzCalcLength(M,I)   \tkzGetLength{dMI}
\tkzFillPolygon[color = orange](A,B,C,D)
\tkzFillCircle[R,color = yellow](M,\dMI pt)
\tkzFillCircle[R,color = blue!50!black](F,4 cm)%
\end{tikzpicture}
```

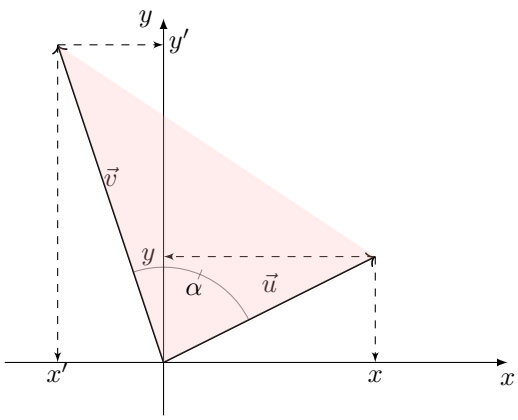
11.9 多边形着色

\tkzFillPolygon[选项](点集列表)

可以在对多边形着色，但该命令仅对内部着色，不绘制多边形

参数	样例	说明
$(pt1, pt2, pt3, \dots)$	\tkzFillPolygon(A,B,C)	对多边形着色
选项	**默认值**	**样例**
TikZ 选项	无	\tkzFillPolygon[red](A,B,C)

11.9.1 tkzFillPolygon 命令示例




```
\begin{tikzpicture}[scale=0.7]
\tkzInit[xmin=-3,xmax=6,ymin=-1,ymax=6]
\tkzDrawX[noticks] \tkzDrawY[noticks]
\tkzDefPoints{0/0/O,4/2/A,-2/6/B}
\tkzPointShowCoord[xlabel=$x$,ylabel=$y$](A)
\tkzPointShowCoord[xlabel=$x'$,ylabel=$y'$,ystyle={right=2pt}](B)
\tkzDrawSegments[->](O,A O,B)
\tkzLabelSegment[above=3pt](O,A){$\vec{u}$}
\tkzLabelSegment[above=3pt](O,B){$\vec{v}$}
\tkzMarkAngle[fill= yellow,size=1.8cm,opacity=.5](A,O,B)
\tkzFillPolygon[red!30,opacity=0.25](A,B,O)
\tkzLabelAngle[pos = 1.5](A,O,B){$\alpha$}
\end{tikzpicture}
```

11.10 定义正多边形

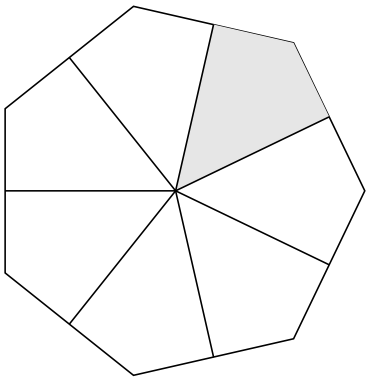
\tkzDefRegPolygon[选项]($pt1, pt2$)

根据选项中指定的边数，以指定的点为中心或是指定的边，定义一个正多边形

参数	样例	说明
$pt1, pt2$	(O, A)	如果使用 “center” 选项，则 O 是多边形中心，A 是多边形一个顶点
$pt1, pt2$	(A, B)	如果使用 “side” 选项，则 $[AB]$ 是多边形一条边

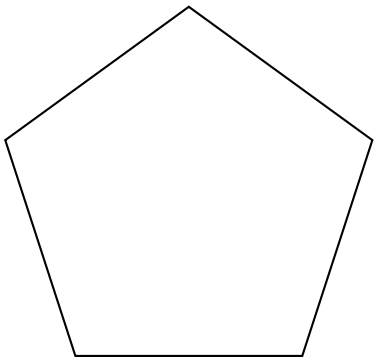
选项	默认值	样例
name	P	顶点命名为 $P1, P2, \ldots$
sides	5	边数
center	center	第 1 个点是正多边形中心
side	center	指定的两个顶点构成一条边
TikZ 选项	...	

11.10.1 center 选项示例



```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints{0/0/P0,0/0/Q0,2/0/P1}
\tkzDefMidPoint(P0,P1) \tkzGetPoint{Q1}
\tkzDefRegPolygon[center,sides=7](P0,P1)
\tkzDefMidPoint(P1,P2) \tkzGetPoint{Q1}
\tkzDefRegPolygon[center,sides=7,name=Q](P0,Q1)
\tkzDrawPolygon(P1,P...,P7)
\tkzFillPolygon[gray!20](Q0,Q1,P2,Q2)
\foreach \j in {1,...,7} {
\tkzDrawSegment[black](P0,Q\j)}
\end{tikzpicture}
```

11.10.2 side 选示例



```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{-4/0/A, -1/0/B}
\tkzDefRegPolygon[side,sides=5,name=P](A,B)
\tkzDrawPolygon[thick](P1,P...,P5)
\end{tikzpicture}
```

第十二章 圆

通过本节的命令中，可以定义并绘制圆。为此，需要知道圆心以及半径或圆上的点。常用的方法是给定圆心绘制过指定的点的圆，这是默认方法，否则则需要给出圆的半径 **R**。另外，还有一些特殊的圆，例如三角形的外接圆等。

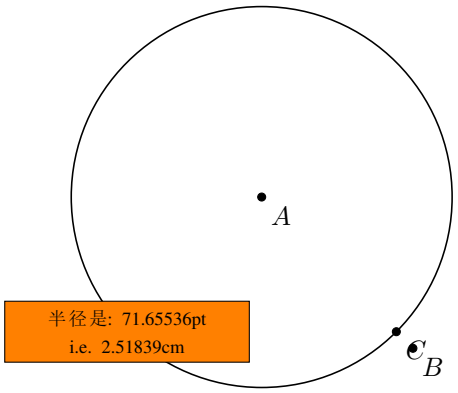
- tkzDefCircle 命令根据指定的圆心和半径 (单位：cm) 定义一个圆，\tkzGetPoint 和\tkzGetLength 命令得到圆心和半径，或使用\tkzPointResult 命令和\tkzLengthResult 命令使用这些值，但不命名;
- tkzDrawCircle 命令用于绘制圆;
- tkzFillCircle 命令用于在不绘制圆的情况下对圆进行着色;
- tkzClipCircle 命令用于用圆进行裁剪;
- tkzLabelCircle 命令用于标注一个圆.

12.1 定义圆

\tkzDefCircle[选项](A,B) or (A,B,C)

参数	样例	说明
(A,B) or (A,B,C)	(A,B)	$[A,B]$ 是半径，A 是圆心
选项	**默认值**	**含义**
through	through	两点间的距离是半径
diameter	through	两点间的距离是直径
circum	through	三角形的外接圆
in	through	三角形的内切圆
ex	through	三角形的旁切圆
euler or nine	through	三角形的欧拉圆
spieker	through	三角形的 Spieker 圆 (斯俾克圆)
orthogonal	through	与指定圆心的另一个圆正交
orthogonal through	through	与通过两个点的另一个圆正交
apollonius	through	Apollonius 圆 (阿波罗尼圆、阿氏圆)
K	1	Apollonius 圆的系数

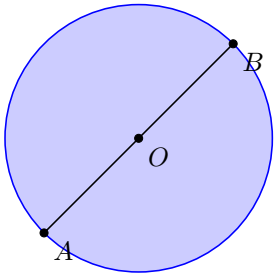
12.1.1 使用随机点与 through 选项示例




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,4){A} \tkzDefPoint(2,2){B}
\tkzDefMidPoint(A,B) \tkzGetPoint{I}
\tkzDefRandPointOn[segment = I-B] \tkzGetPoint{C}
\tkzDefCircle[through](A,C)
\tkzGetLength{rACpt}
\tkzpttocm(\rACpt){rACcm}
\tkzDrawCircle(A,C)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,B,C)
\tkzLabelCircle[draw,fill=orange, text width=3cm,
text centered, font=\scriptsize](A,C)(-90)%
{半径是:  \rACpt pt i.e. \rACcm cm}
\end{tikzpicture}
```

12.1.2 diameter 选项示例

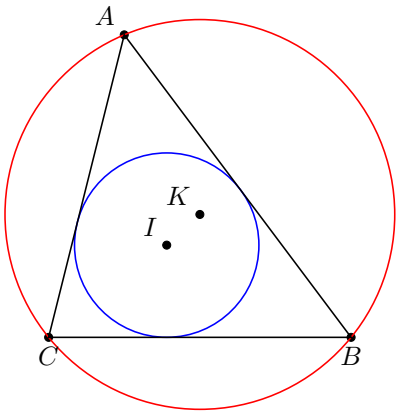
可以通过 $[AB]$ 的中点确定圆心。




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,2){B}
\tkzDefCircle[diameter](A,B)
\tkzGetPoint{O}
\tkzDrawCircle[blue,fill=blue!20](O,B)
\tkzDrawSegment(A,B)
\tkzDrawPoints(A,B,O)
\tkzLabelPoints(A,B,O)
\end{tikzpicture}
```

12.1.3 三角形的内切圆和外接圆示例

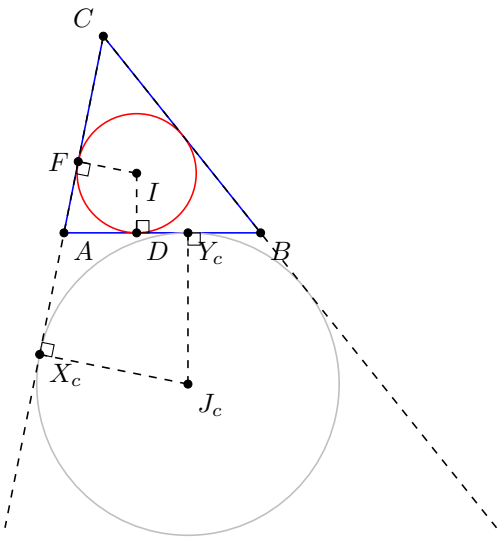
可以使用\tkzGetFirstPoint{I} 和\tkzGetSecondPoint{Ib} 命令得到内切圆在对应边上的投影。




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(2,2){A}
\tkzDefPoint(5,-2){B}
\tkzDefPoint(1,-2){C}
\tkzDefCircle[in](A,B,C)
\tkzGetPoint{I} \tkzGetLength{rIN}
\tkzDefCircle[circum](A,B,C)
\tkzGetPoint{K} \tkzGetLength{rCI}
\tkzDrawPoints(A,B,C,I,K)
\tkzDrawCircle[R,blue](I,\rIN pt)
\tkzDrawCircle[R,red](K,\rCI pt)
\tkzLabelPoints[below](B,C)
\tkzLabelPoints[above left](A,I,K)
\tkzDrawPolygon(A,B,C)
\end{tikzpicture}
```

12.1.4 ex 选项示例

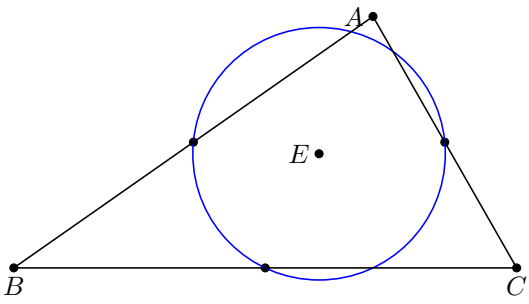
与顶点 C 对应的旁切圆。




```
\begin{tikzpicture}[scale=.65]
\tkzDefPoints{ 0/0/A,4/0/B,0.8/4/C}
\tkzDefCircle[ex](B,C,A)
\tkzGetPoint{J_c} \tkzGetLength{rc}
\tkzDefPointBy[projection=onto A-C ](J_c) \tkzGetPoint{X_c}
\tkzDefPointBy[projection=onto A-B ](J_c) \tkzGetPoint{Y_c}
\tkzGetPoint{I}% 内切圆圆心
% 感觉是\tkzDefCircle[ex](B,C,A) 命令同时定义出了旁切园和
内切圆的圆心
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawCircle[R,color=lightgray](J_c,\rc pt)
% possible  \tkzDrawCircle[ex](A,C,B)
\tkzDrawCircle[in,color=red](A,B,C)    \tkzGetPoint{I}
\tkzDefPointBy[projection=onto A-C ](I) \tkzGetPoint{F}
\tkzDefPointBy[projection=onto A-B ](I) \tkzGetPoint{D}
\tkzDrawLines[add=0 and 1.5,dashed](C,A C,B)
\tkzDrawSegments[dashed](J_c,X_c I,D  I,F J_c,Y_c)
\tkzMarkRightAngles(A,F,I B,D,I J_c,X_c,A J_c,Y_c,B)
\tkzDrawPoints(B,C,A,I,D,F,X_c,J_c,Y_c)
\tkzLabelPoints(B,A,J_c,I,D,X_c,Y_c)
\tkzLabelPoints[above left](C)
\tkzLabelPoints[left](F)
\end{tikzpicture}
```

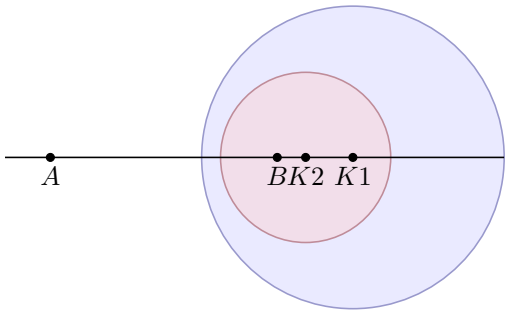
12.1.5 euler 选项示例

同时验证了欧拉圆会通过三角形三个边的中点。




```
\begin{tikzpicture}[scale=.95]
\tkzDefPoint(5,3.5){A}
\tkzDefPoint(0,0){B}
\tkzDefPoint(7,0){C}
\tkzDefCircle[euler](A,B,C)
\tkzGetPoint{E}
\tkzGetLength{rEuler}
\tkzDefSpcTriangle[medial](A,B,C){M_a,M_b,M_c}
\tkzDrawPoints(A,B,C,E,M_a,M_b,M_c)
\tkzDrawCircle[R,blue](E,\rEuler pt)
\tkzDrawPolygon(A,B,C)
\tkzLabelPoints[below](B,C)
\tkzLabelPoints[left](A,E)
\end{tikzpicture}
```

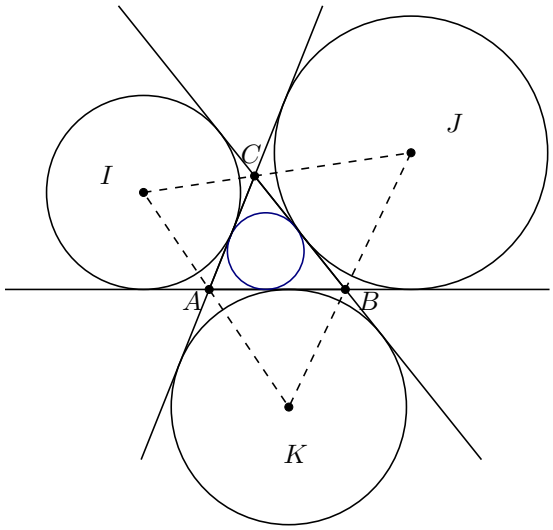
12.1.6 apollonius 选项示例




```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDefCircle[apollonius,K=2](A,B)
\tkzGetPoint{K1}
\tkzGetLength{rAp}
\tkzDrawCircle[R,color = blue!50!black,
fill=blue!20,opacity=.4](K1,\rAp pt)
\tkzDefCircle[apollonius,K=3](A,B)
\tkzGetPoint{K2}   \tkzGetLength{rAp}
\tkzDrawCircle[R,color=red!50!black,
fill=red!20,opacity=.4](K2,\rAp pt)
\tkzLabelPoints[below](A,B,K1,K2)
\tkzDrawPoints(A,B,K1,K2)
\tkzDrawLine[add=.2 and 1](A,B)
\end{tikzpicture}
```

12.1.7 ex 选项示例

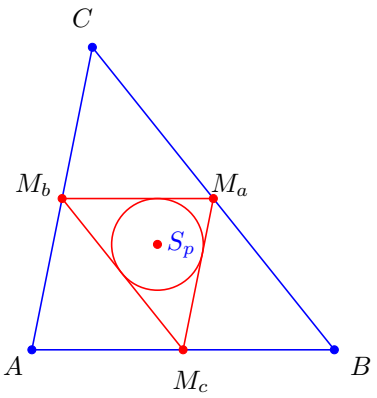
可以使用\tkzGetFirstPoint{Jb} 和\tkzGetSecondPoint{Tb} 命令，得到旁切圆圆心在边上的投影。




```
\begin{tikzpicture}[scale=.6]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefPoint(1,2.5){C}
\tkzDefCircle[ex](A,B,C) \tkzGetPoint{I}
\tkzGetLength{rI}
\tkzDefCircle[ex](C,A,B) \tkzGetPoint{J}
\tkzGetLength{rJ}
\tkzDefCircle[ex](B,C,A) \tkzGetPoint{K}
\tkzGetLength{rK}
\tkzDefCircle[in](B,C,A) \tkzGetPoint{O}
\tkzGetLength{rO}
\tkzDrawLines[add=1.5 and 1.5](A,B A,C B,C)
\tkzDrawPoints(I,J,K)
\tkzDrawPolygon(A,B,C)
\tkzDrawPolygon[dashed](I,J,K)
\tkzDrawCircle[R,blue!50!black](O,\rO)
%\tkzDrawSegments[dashed](A,K B,J C,I)
\tkzDrawPoints(A,B,C)
\tkzDrawCircles[R](J,{\rJ} I,{\rI} K,{\rK})
\tkzAutoLabelPoints[center=O,dist=.3](A,B,C,I,J,K)
\end{tikzpicture}
```

12.1.8 spieker 选项示例

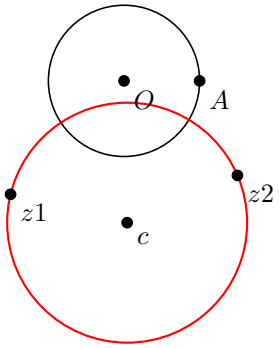
三角形三个边的中点构成的三角形 $M_aM_bM_c$ 的内切圆是 Spieker 圆:




```
\begin{tikzpicture}[scale=1]
\tkzDefPoints{ 0/0/A,4/0/B,0.8/4/C}
\tkzDefSpcTriangle[medial](A,B,C){M_a,M_b,M_c}
\tkzDefTriangleCenter[spieker](A,B,C) \tkzGetPoint{S_p}
\tkzDrawPolygon[blue](A,B,C)
\tkzDrawPolygon[red](M_a,M_b,M_c)
\tkzDrawPoints[blue](B,C,A)
\tkzDrawPoints[red](M_a,M_b,M_c,S_p)
\tkzDrawCircle[in,red](M_a,M_b,M_c)
\tkzAutoLabelPoints[center=S_p,dist=.3](M_a,M_b,M_c)
\tkzLabelPoints[blue,right](S_p)
\tkzAutoLabelPoints[center=S_p](A,B,C)
\end{tikzpicture}
```

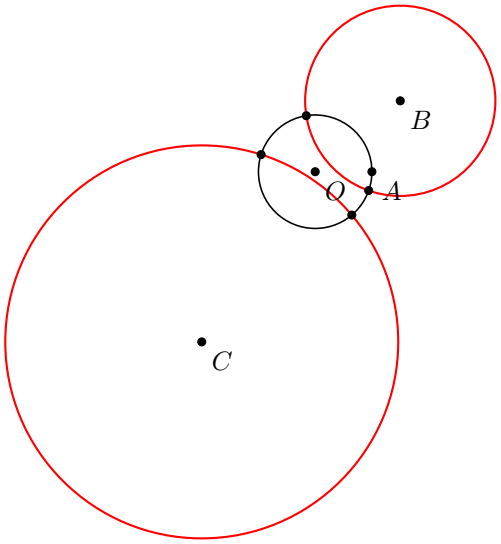
12.1.9 orthogonal through 选项示例

过指定两个点的另一个圆的正交圆。




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,0){O} \tkzDefPoint(1,0){A}
\tkzDrawCircle(O,A)
\tkzDefPoint(-1.5,-1.5){z1} \tkzDefPoint(1.5,-1.25){z2}
\tkzDefCircle[orthogonal through=z1 and z2](O,A)
\tkzGetPoint{c}
\tkzDrawCircle[thick,color=red](tkzPointResult,z1)
\tkzDrawPoints[fill=red,color=black,size=4](O,A,z1,z2,c)
\tkzLabelPoints(O,A,z1,z2,c)
\end{tikzpicture}
```

12.1.10 指定圆心的另一个圆的正交圆示例




```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/O,1/0/A}
\tkzDefPoints{1.5/1.25/B,-2/-3/C}
\tkzDefCircle[orthogonal from=B](O,A)
\tkzGetPoints{z1}{z2}
\tkzDefCircle[orthogonal from=C](O,A)
\tkzGetPoints{t1}{t2}
\tkzDrawCircle(O,A)
\tkzDrawCircle[thick,color=red](B,z1)
\tkzDrawCircle[thick,color=red](C,t1)
\tkzDrawPoints(t1,t2,C)
\tkzDrawPoints(z1,z2,O,A,B)
\tkzLabelPoints(O,A,B,C)
\end{tikzpicture}
```

12.2 绘制圆

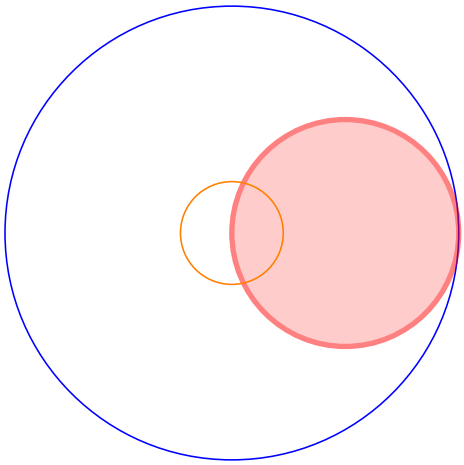
\tkzDrawCircle[选项](A,B)

只能用两个点指定半径或直径，使用 R 选项，则需要直接指定半径

参数	样例	说明
$(pt1,pt2)$	(A,B)	两个点定义半径或直径
选项	**默认值**	**定义**
through	through	两个点定义半径
diameter	through	两个点定义直径
R	through	需要指定圆心、半径

12.2.1 绘制一个圆并对其进行着色

能够在绘制中对圆进行着色。



```
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(3,0){A}
% 圆心是 O，通过 A 点
\tkzDrawCircle[color=blue](O,A)
% 直径是 $[OA]$
\tkzDrawCircle[diameter,color=red,%
line width=2pt,fill=red!40,opacity=.5](O,A)
% 圆心是 O，半径 =exp(1) cm
\edef\rayon{\fpeval{0.25*exp(1)}}
\tkzDrawCircle[R,color=orange](O,\rayon cm)
\end{tikzpicture}
```

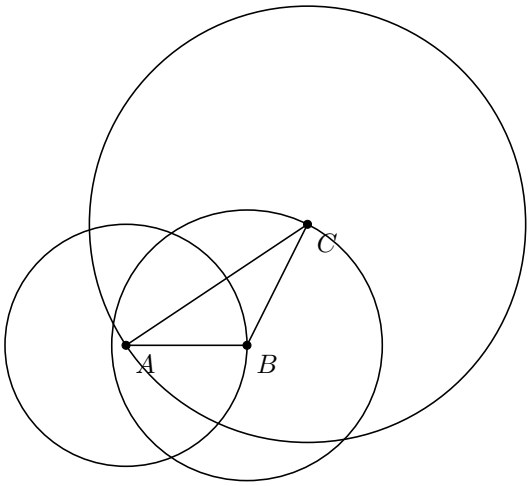
12.3 绘制多个圆

\tkzDrawCircles[选项]$(A,B\quad C,D)$

参数是空格分隔的构成圆的点对列表，点对中的两个点之间用逗号分隔。使用 R 选项，则需要直接指定半径

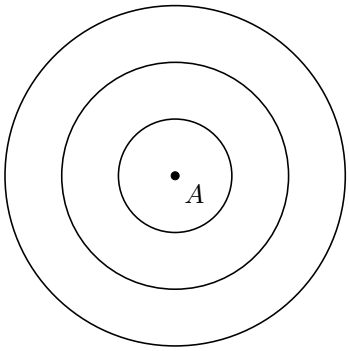
参数	样例	说明
$(pt1,pt2\quad pt3,pt4\ \ldots)$	$(A,B\quad C,D)$	点集列表
选项	**默认值**	**定义**
through	through	两个点定义半径
diameter	through	两个点定义直径
R	through	需要指定圆心、半径

12.3.1 通过三角形定义圆示例



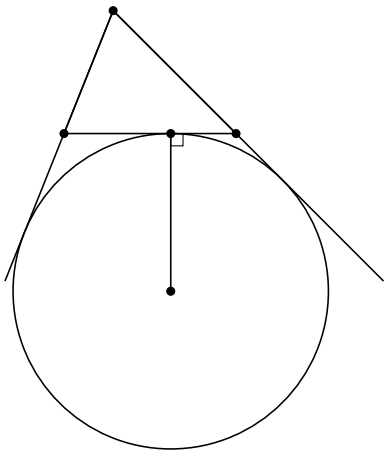

```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,0){B}
\tkzDefPoint(3,2){C}
\tkzDrawPolygon(A,B,C)
\tkzDrawCircles(A,B B,C C,A)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,B,C)
\end{tikzpicture}
```

12.3.2 同心圆示例




```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoint(0,0){A}
\tkzDrawCircles[R](A,1cm A,2cm A,3cm)
\tkzDrawPoint(A)
\tkzLabelPoints(A)
\end{tikzpicture}
```

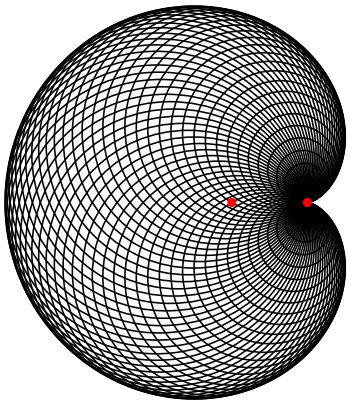
12.3.3 旁切圆示例



```
\begin{tikzpicture}[scale=0.65]
\tkzDefPoints{0/0/A,3.5/0/B,1/2.5/C}
\tkzDrawPolygon(A,B,C)
\tkzDefCircle[ex](B,C,A)
\tkzGetPoint{J_c} \tkzGetSecondPoint{T_c}
\tkzGetLength{rJc}
\tkzDrawCircle[R](J_c,{\rJc pt})
\tkzDrawLines[add=0 and 1.2](C,A C,B)
\tkzDrawSegment(J_c,T_c)
\tkzMarkRightAngle(J_c,T_c,B)
\tkzDrawPoints(A,B,C,J_c,T_c)
\end{tikzpicture}
```

12.3.4 心形线示例

名称来源于希腊语中的 *kardia (heart)*，是根据其形状命名的。这个名称最先是由 Johan Castillon 给出的 (Wikipedia)。

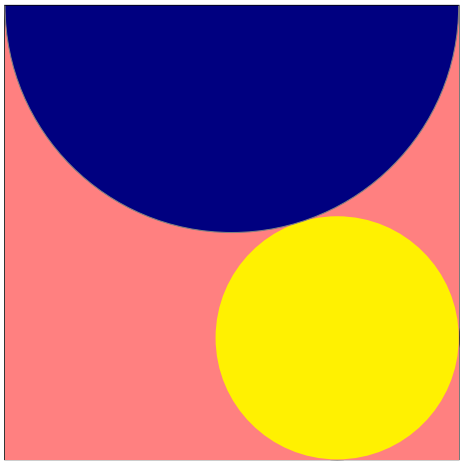


```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,0){A}
\foreach \ang in {5,10,...,360}{%
\tkzDefPoint(\ang:2){M}
\tkzDrawCircle(M,A)}
\tkzDrawPoints[red](O,A)
\end{tikzpicture}
```

12.4 绘制半圆

\tkzDrawSemiCircle[选项](A,B)		
A,B 的顺序将影响半圆的方向		
参数	**样例**	**说明**
$(pt1,pt2)$	(A,B)	两个点定义半径或直径
选项	**默认值**	**定义**
through	through	两个点定义半径
diameter	through	两个点定义直径

12.4.1 示例



```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(6,0){B}
\tkzDefSquare(A,B) \tkzGetPoints{C}{D}
\tkzDrawPolygon(B,C,D,A)
\tkzDefPoint(3,6){F}
\tkzDefTriangle[equilateral](C,D) \tkzGetPoint{I}
\tkzDefPointBy[projection=onto B-C](I) \tkzGetPoint{J}
\tkzInterLL(D,B)(I,J)  \tkzGetPoint{K}
\tkzDefPointBy[symmetry=center K](B) \tkzGetPoint{M}
\tkzDrawCircle(M,I)
\tkzCalcLength(M,I)  \tkzGetLength{dMI}
\tkzFillPolygon[color = red!50](A,B,C,D)
\tkzFillCircle[R,color = yellow](M,\dMI pt)
\tkzDrawSemiCircle[fill = blue!50!black](F,D)%
\end{tikzpicture}
```

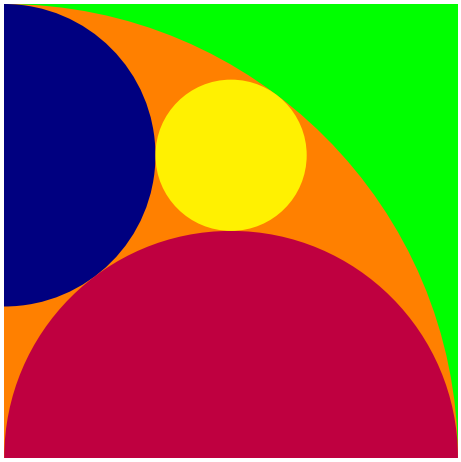
12.5 给圆着色

在绘制圆时，也可以实现着色，但该命令不绘制圆，仅对圆形区域进行着色。可以使用所有有效 TikZ 样式

\tkzFillCircle[选项](A, B)

选项	默认值	定义
radius	radius	两个点定义半径
R	radius	圆心、半径模式着色圆

12.5.1 sangaku 圆示例



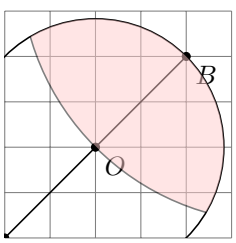
```
\begin{tikzpicture}
\tkzInit[xmin=0,xmax = 6,ymin=0,ymax=6]
\tkzDefPoint(0,0){B}  \tkzDefPoint(6,0){C}%
\tkzDefSquare(B,C)    \tkzGetPoints{D}{A}
\tkzClipPolygon(B,C,D,A)
\tkzDefMidPoint(A,D)  \tkzGetPoint{F}
\tkzDefMidPoint(B,C)  \tkzGetPoint{E}
\tkzDefMidPoint(B,D)  \tkzGetPoint{Q}
\tkzDefTangent[from = B](F,A) \tkzGetPoints{G}{H}
\tkzInterLL(F,G)(C,D) \tkzGetPoint{J}
\tkzInterLL(A,J)(F,E) \tkzGetPoint{K}
\tkzDefPointBy[projection=onto B-A](K) \tkzGetPoint{M}
\tkzFillPolygon[color = green](A,B,C,D)
\tkzFillCircle[color = orange](B,A)
\tkzFillCircle[color = blue!50!black](M,A)
\tkzFillCircle[color = purple](E,B)
\tkzFillCircle[color = yellow](K,Q)
\end{tikzpicture}
```

12.6 用圆裁剪

\tkzClipCircle[选项](A, B) 或 (A, r)

参数	样例	说明
(A, B) 或 (A, r)	(A, B) 或 $(A, 2cm)$	AB 是半径或直径
选项	**默认值**	**含义**
radius	radius	两个点确定半径
R	radius	指定半径

12.6.1 示例



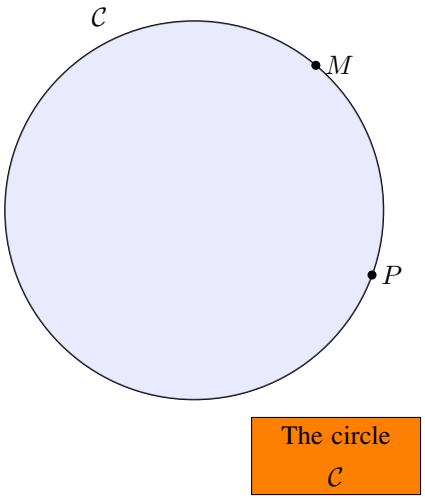

```
\begin{tikzpicture}[scale=.6]
\tkzInit[xmax=5,ymax=5] \tkzGrid \tkzClip
\tkzDefPoint(0,0){A} \tkzDefPoint(2,2){O}
\tkzDefPoint(4,4){B} \tkzDefPoint(6,6){C}
\tkzDrawPoints(O,A,B,C) \tkzLabelPoints(O,A,B,C)
\tkzDrawCircle(O,A) \tkzClipCircle(O,A)
\tkzDrawLine(A,C) \tkzDrawCircle[fill=red!20,opacity=.5](C,O)
\end{tikzpicture}
```

12.7 为圆添加标注

\tkzLabelCircle[选项](A, B)(角度){标注}

选项	默认值	含义
radius	radius	两个点确定半径
R	radius	指定半径
可以使用所有有效 TikZ 样式，标注内容通过“传递”给大括号中的的参数指定		

12.7.1 标注示例




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,0){N}
\tkzDefPointBy[rotation=center O angle 50](N)
\tkzGetPoint{M}
\tkzDefPointBy[rotation=center O angle -20](N)
\tkzGetPoint{P}
\tkzDefPointBy[rotation=center O angle 125](N)
\tkzGetPoint{P'}
\tkzLabelCircle[above=4pt](O,N)(120){$\mathcal{C}$}
\tkzDrawCircle(O,M)
\tkzFillCircle[color=blue!20,opacity=.4](O,M)
\tkzLabelCircle[R,draw,fill=orange, text width=2cm,
text centered](O,3 cm)(-60)%
{The circle\\ $\mathcal{C}$}
\tkzDrawPoints(M,P)
\tkzLabelPoints[right](M,P)
\end{tikzpicture}
```

第十三章 交点

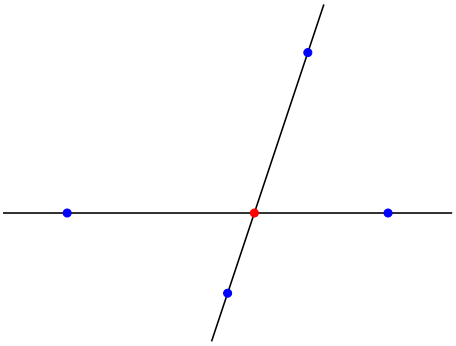
可以求得两条直线、一条直线与一个圆及两个圆之间的交点。求交点的相关命令没有可选参数，用户必须确保交点存在。

13.1 两条直线的交点

\tkzInterLL$(A,B)(C,D)$

求直线 (A,B) 和 (C,D) 的交点，并保存于\tkzPointResult 命令中，两条直线分别由两个圆括号中的点对定义。可以通过\tkzDefPoint 保存并命令交点

13.1.1 直线交点示例



```
\begin{tikzpicture}[rotate=-45,scale=.75]
\tkzDefPoint(2,1){A} \tkzDefPoint(6,5){B}
\tkzDefPoint(3,6){C} \tkzDefPoint(5,2){D}
\tkzDrawLines(A,B C,D)
\tkzInterLL(A,B)(C,D)
\tkzGetPoint{I}
\tkzDrawPoints[color=blue](A,B,C,D)
\tkzDrawPoint[color=red](I)
\end{tikzpicture}
```

13.2 一条直线和一个圆的交点

直线可以由两个点定义，圆可以按如下方式进行定义：

- (O,C) O 是圆心，C 是圆上的一个点。
- (O,r) O 是圆心，r 半径，单位可以是 cm 或 pt

\tkzInterLC[选项]$(A,B)(O,C)$ 或 (O,r) 或 (O,C,D)

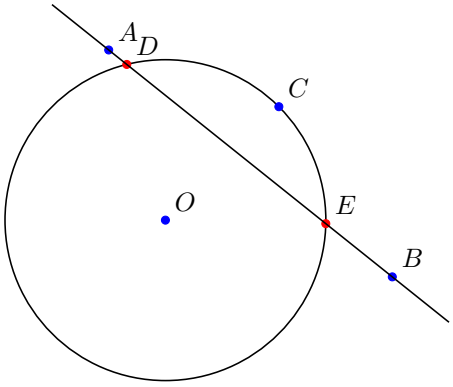
参数必须是一条直线和一个圆

选项	默认值	含义
N	N	(O,C)
R	N	(O, 1 cm) 或 (O, 120 pt)
with nodes	N	(O,C,D)CD 是半径

定义直线与由圆心 O 和半径 r 定义的圆的交点 I 和 J，如果出现错误，则记录在.log 日志文件中

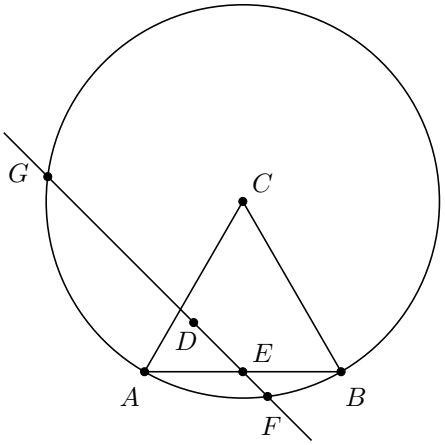
13.2.1 直线与圆的交点示例

在下面示例代码中，圆用两个点表示，直线与圆的有两个交点。




```
\begin{tikzpicture}[scale=.75]
\tkzInit[xmax=5,ymax=4]
\tkzDefPoint(1,1){O} \tkzDefPoint(0,4){A}
\tkzDefPoint(5,0){B} \tkzDefPoint(3,3){C}
\tkzInterLC(A,B)(O,C)  \tkzGetPoints{D}{E}
\tkzDrawCircle(O,C)
\tkzDrawPoints[color=blue](O,A,B,C)
\tkzDrawPoints[color=red](D,E)
\tkzDrawLine(A,B)
\tkzLabelPoints[above right](O,A,B,C,D,E)
\end{tikzpicture}
```

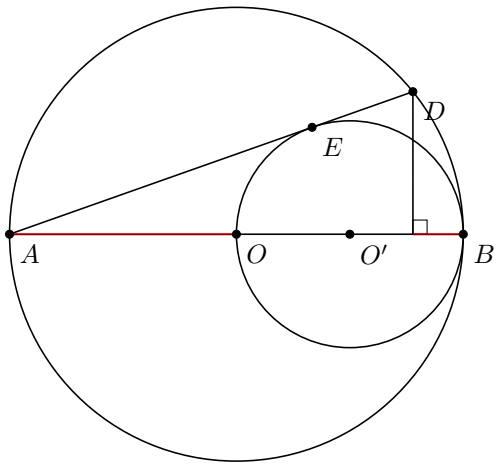
13.2.2 “with nodes” 选项示例




```
\begin{tikzpicture}[scale=.65]
\tkzDefPoints{0/0/A,4/0/B,1/1/D,2/0/E}
\tkzDefTriangle[equilateral](A,B) \tkzGetPoint{C}
\tkzDrawCircle(C,A)
\tkzInterLC[with nodes](D,E)(C,A,B)
\tkzGetPoints{F}{G}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,...,G)
\tkzDrawLine(F,G)
\tkzAutoLabelPoints[center=C](A,B,D,F,G)
\tkzLabelPoints[above right](C,E)
\end{tikzpicture}
```

13.2.3 直线与圆的交点复杂示例

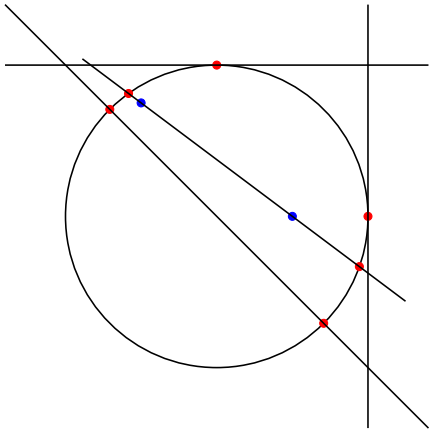
可参阅http://www.gogeometry.com/problem/p190_tangent_circle_diameter_perpendicular.htm




```
\begin{tikzpicture}[scale=.75]
\tkzDefPoint(0,0){A} \tkzDefPoint(8,0){B}
\tkzDefMidPoint(A,B) \tkzGetPoint{O}
\tkzDrawCircle(O,B)
\tkzDefMidPoint(O,B) \tkzGetPoint{O'}
\tkzDrawCircle(O',B)
\tkzDefTangent[from=A](O',B) \tkzGetSecondPoint{E}
\tkzInterLC(A,E)(O,B)
\tkzGetSecondPoint{D}
\tkzDefPointBy[projection=onto A-B](D) \tkzGetPoint{F}
\tkzMarkRightAngle(D,F,B)
\tkzDrawSegments(A,D A,B D,F)
\tkzDrawSegments[color=red,line width=1pt,opacity=.5](A,O F,B)
\tkzDrawPoints(A,B,O,O',E,D)
\tkzLabelPoints(A,B,O,O',E,D)
\end{tikzpicture}
```

13.2.4 由圆心和半径定义圆示例

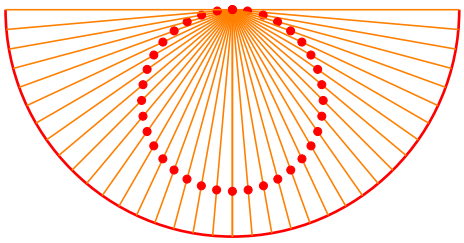
直线与圆相切的特例：



```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,8){A}  \tkzDefPoint(8,0){B}
\tkzDefPoint(8,8){C}  \tkzDefPoint(4,4){I}
\tkzDefPoint(2,7){E}  \tkzDefPoint(6,4){F}
\tkzDrawCircle[R](I,4 cm)
\tkzInterLC[R](A,C)(I,4 cm)  \tkzGetPoints{I1}{I2}
\tkzInterLC[R](B,C)(I,4 cm)  \tkzGetPoints{J1}{J2}
\tkzInterLC[R](A,B)(I,4 cm)  \tkzGetPoints{K1}{K2}
\tkzDrawPoints[color=red](I1,J1,K1,K2)
\tkzDrawLines(A,B B,C A,C)
\tkzInterLC[R](E,F)(I,4 cm)  \tkzGetPoints{I2}{J2}
\tkzDrawPoints[color=blue](E,F)
\tkzDrawPoints[color=red](I2,J2)
\tkzDrawLine(I2,J2)
\end{tikzpicture}
```

13.2.5 更为复杂的示例

注意语法细节：首先，在传递参数的同时，可以计算点的坐标，但是必须嵌套 xfp 语法。由于 xfp 宏包能够使用弧度，如使用 pi，当然，也可以使用度，但需要使用类似 sind 或 cosd 命令进行计算。其次，当计算中需要圆括号时，需要使用分组命令：...TEX{...}.



```
\begin{tikzpicture}[scale=1.2]
\tkzDefPoint(0,1){J}  \tkzDefPoint(0,0){O}
\tkzDrawArc[R,line width=1pt,color=red](J,2.5 cm)(180,0)
\foreach \i in {0,-5,-10,...,-85,-90}{
\tkzDefPoint({2.5*cosd(\i)},{1+2.5*sind(\i)}){P}
\tkzDrawSegment[color=orange](J,P)
\tkzInterLC[R](P,J)(O,1 cm)
\tkzGetPoints{M}{N}
\tkzDrawPoints[red](N)
}
\foreach \i in {-90,-95,...,-175,-180}{
\tkzDefPoint({2.5*cosd(\i)},{1+2.5*sind(\i)}){P}
\tkzDrawSegment[color=orange](J,P)
\tkzInterLC[R](P,J)(O,1 cm)
\tkzGetPoints{M}{N}
\tkzDrawPoints[red](M)
}
\end{tikzpicture}
```

13.3 求两个圆的交点

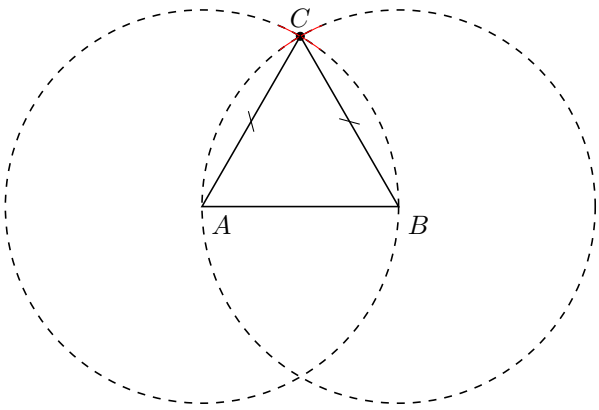
通常，圆是由圆心和另一个点确定的，但也可以用 R 选项后，在参数中指定半径

\tkzInterCC[选项]$(O,A)(O',A')$ 或 $(O,r)(O',r')$ 或 $(O,A,B)(O',C,D)$

选项	默认值	含义
N	N	OA 和 $O'A'$ 是半径, O 和 O' 是圆心
R	N	r 和 r' 是半径
with nodes	N	在 $(O,A,B)(O',C,D)$ 中 AB 和 CD 是半径

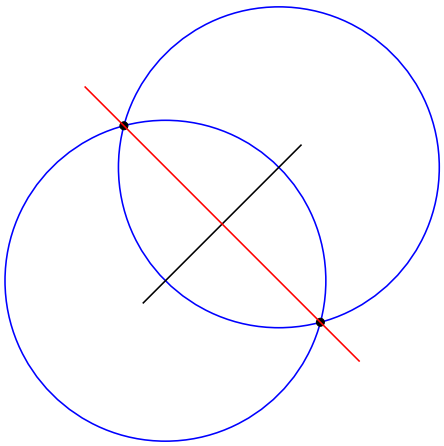
该命令定义了 O 和 O' 两个圆的交点，如果两个圆没有交点，则返回错误。
也可以直接使用\tkzInterCCN 命令和\tkzInterCCR 命令进行计算

13.3.1 构造等边三角形示例



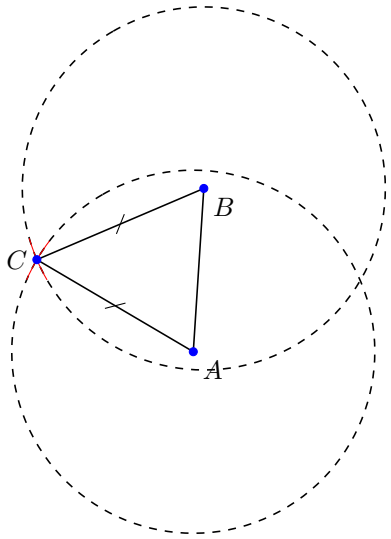

```
\begin{tikzpicture}[trim left=-1cm,scale=0.65]
\tkzDefPoint(1,1){A}
\tkzDefPoint(5,1){B}
\tkzInterCC(A,B)(B,A)\tkzGetPoints{C}{D}
\tkzDrawPoint[color=black](C)
\tkzDrawCircle[dashed](A,B)
\tkzDrawCircle[dashed](B,A)
\tkzCompass[color=red](A,C)
\tkzCompass[color=red](B,C)
\tkzDrawPolygon(A,B,C)
\tkzMarkSegments[mark=s|](A,C B,C)
\tkzLabelPoints[](A,B)
\tkzLabelPoint[above](C){$C$}
\end{tikzpicture}
```

13.3.2 求中点示例



```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(2,2){B}
\tkzDrawCircle[color=blue](B,A)
\tkzDrawCircle[color=blue](A,B)
\tkzInterCC(B,A)(A,B)\tkzGetPoints{M}{N}
\tkzDrawLine(A,B)
\tkzDrawPoints(M,N)
\tkzDrawLine[color=red](M,N)
\end{tikzpicture}
```

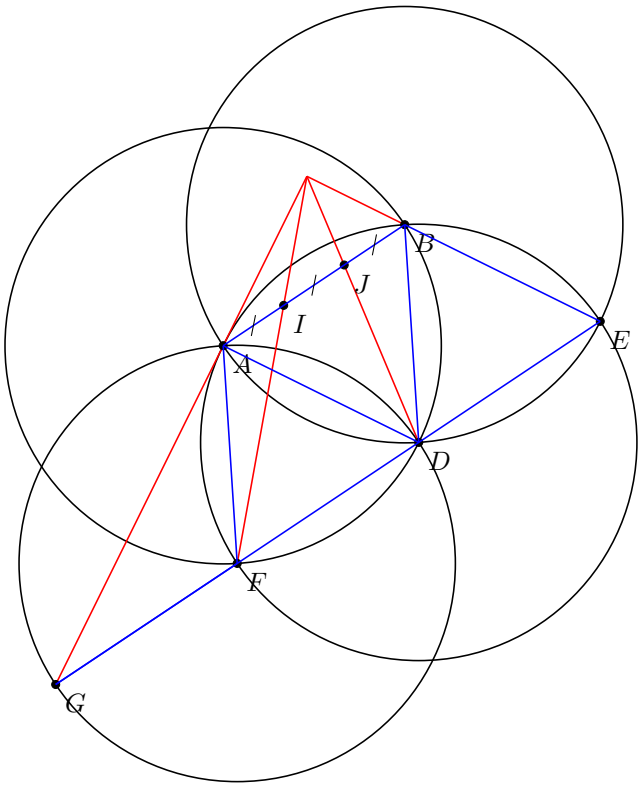
13.3.3 求等腰三角形示例




```
\begin{tikzpicture}[rotate=120,scale=0.6]
\tkzDefPoint(1,2){A}
\tkzDefPoint(4,0){B}
\tkzInterCC[R](A,4cm)(B,4cm)
\tkzGetPoints{C}{D}
\tkzDrawCircle[R,dashed](A,4 cm)
\tkzDrawCircle[R,dashed](B,4 cm)
\tkzCompass[color=red](A,C)
\tkzCompass[color=red](B,C)
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints[color=blue](A,B,C)
\tkzMarkSegments[mark=s|](A,C B,C)
\tkzLabelPoints[](A,B)
\tkzLabelPoint[left](C){$C$}
\end{tikzpicture}
```

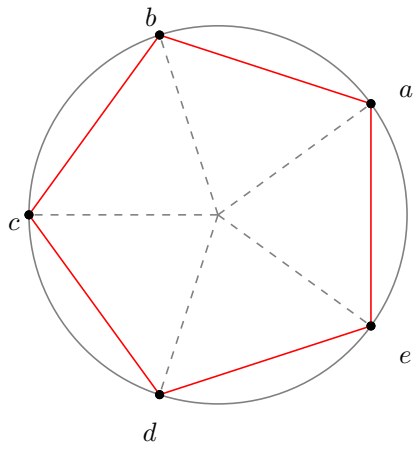
13.3.4 三等分线段示例

与尺规作图方式将线段三等分操作一致。




```
\begin{tikzpicture}[scale=0.80]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,2){B}
\tkzInterCC(A,B)(B,A)
\tkzGetPoints{C}{D}
\tkzInterCC(D,B)(B,A)
\tkzGetPoints{A}{E}
\tkzInterCC(D,B)(A,B)
\tkzGetPoints{F}{B}
\tkzInterLC(E,F)(F,A)
\tkzGetPoints{D}{G}
\tkzInterLL(A,G)(B,E)
\tkzGetPoint{O}
\tkzInterLL(O,D)(A,B)
\tkzGetPoint{J}
\tkzInterLL(O,F)(A,B)
\tkzGetPoint{I}
\tkzDrawCircle(D,A)
\tkzDrawCircle(A,B)
\tkzDrawCircle(B,A)
\tkzDrawCircle(F,A)
\tkzDrawSegments[color=red](O,G O,B O,D O,F)
\tkzDrawPoints(A,B,D,E,F,G,I,J)
\tkzLabelPoints(A,B,D,E,F,G,I,J)
\tkzDrawSegments[blue](A,B B,D A,D A,F F,G E,G B,E)
\tkzMarkSegments[mark=s|](A,I I,J J,B)
\end{tikzpicture}
```

13.3.5 with nodes 选项示例



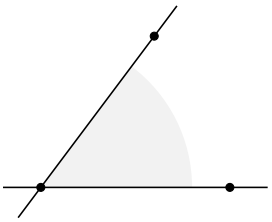

```
\begin{tikzpicture}[scale=.5]
\tkzDefPoints{0/0/A,0/5/B,5/0/C}
\tkzDefPoint(54:5){F}
\tkzDrawCircle[color=gray](A,C)
\tkzInterCC[with nodes](A,A,C)(C,B,F)
\tkzGetPoints{a}{e}
\tkzInterCC(A,C)(a,e) \tkzGetFirstPoint{b}
\tkzInterCC(A,C)(b,a) \tkzGetFirstPoint{c}
\tkzInterCC(A,C)(c,b) \tkzGetFirstPoint{d}
\tkzDrawPoints(a,b,c,d,e)
\tkzDrawPolygon[color=red](a,b,c,d,e)
\foreach \vertex/\num in {a/36,b/108,c/180,
d/252,e/324}{%
\tkzDrawPoint(\vertex)
\tkzLabelPoint[label=\num:$\vertex$](\vertex){}
\tkzDrawSegment[color=gray,style=dashed](A,\vertex)
}
\end{tikzpicture}
```

第十四章 角

14.1 填充角

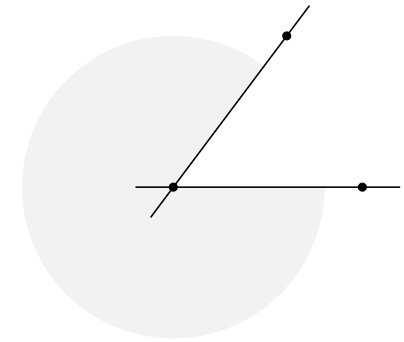
\tkzFillAngle[选项](A,O,B) 复数形式 \tkzFillAngles[选项]$(A,O,B)(A',O',B')$		
O 是角顶点，OA 和 OB 是两条边，点的顺序决定角的方向		
选项	**默认值**	**含义**
size	1 cm	着色扇形的半径
可以使用所有有效的 TikZ 样式，如 fill 和 shade 等		

14.1.1 size 选项示例

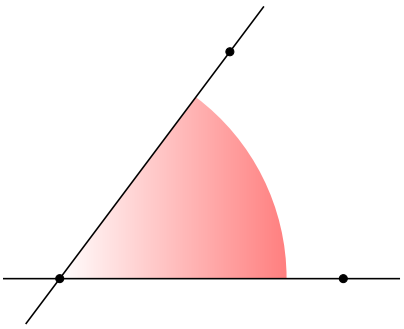


```
\begin{tikzpicture}
\tkzDefPoints{0/0/O,2.5/0/A,1.5/2/B}
\tkzFillAngle[size=2cm, fill=gray!10](A,O,B)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\end{tikzpicture}
```

14.1.2 改变点的顺序示例

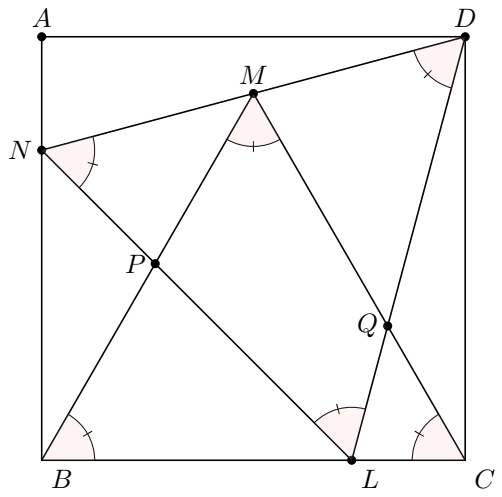


```
\begin{tikzpicture}
\tkzInit
\tkzDefPoints{0/0/O,2.5/0/A,1.5/2/B}
\tkzFillAngle[size=2cm,fill=gray!10](B,O,A)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\end{tikzpicture}
```



```
\begin{tikzpicture}[scale=0.75]
\tkzInit
\tkzDefPoints{0/0/O,5/0/A,3/4/B}
\tkzFillAngle[size=4cm,left color=white,right color=red!50](A,O,B)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\end{tikzpicture}
```

14.1.3 填充多个角示例




```
\begin{tikzpicture}[scale=0.7]
\tkzDefPoints{0/0/B,8/0/C,0/8/A,8/8/D}
\tkzDrawPolygon(B,C,D,A)
\tkzDefTriangle[equilateral](B,C) \tkzGetPoint{M}
\tkzInterLL(D,M)(A,B) \tkzGetPoint{N}
\tkzDefPointBy[rotation=center N angle -60](D) \tkzGetPoint{L}
\tkzInterLL(N,L)(M,B) \tkzGetPoint{P}
\tkzInterLL(M,C)(D,L) \tkzGetPoint{Q}
\tkzDrawSegments(D,N N,L L,D B,M M,C)
\tkzDrawPoints(L,N,P,Q,M,A,D)
\tkzLabelPoints[left](N,P,Q)
\tkzLabelPoints[above](M,A,D)
\tkzLabelPoints(L,B,C)
\tkzMarkAngles(C,B,M B,M,C M,C,B D,L,N L,N,D N,D,L)
\tkzFillAngles[fill=red!20,opacity=.2](C,B,M B,M,C M,C,B D,L,N L,N,D N,D,L)
\end{tikzpicture}
```

14.2 标记角

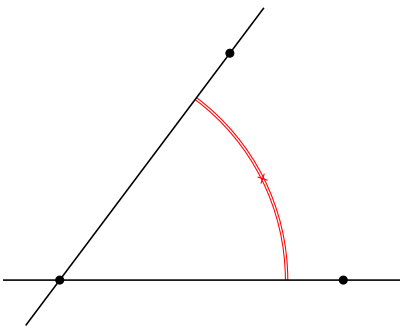
在 TikZ 中，绘图选项非常丰富，本宏包又增加了需要的一些标记，它们定义在 tkz-lib-marks.tex 文件中，主要的标记有:|, ||,|||, z, s, x, o, oo

\tkzMarkAngle[选项](A,O,B) 复数形式 \tkzMarkAngles[选项]$(A,O,B)(A',O',B')$

O 是顶点，注意参数需随选项变化。可以使用任意一种标记，甚至可以绘制一个圆弧，然后为该圆弧添加标记。圆弧的样式通过 arc 选项指定，圆弧的半径由 mksize 选项指定。当然，也可为圆弧着色

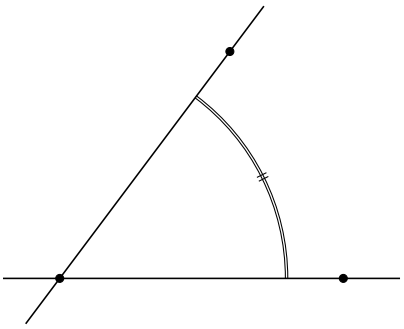
选项	默认值	含义
arc	l	选择单线、双线或三线样式
size	1 cm	圆弧半径
mark	无	标记类型
mksize	4pt	标记符号尺寸
mkcolor	black	标记符号颜色
mkpos	0.5	标记位置

14.2.1 mark = x 选项示例



```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/O,5/0/A,3/4/B}
\tkzMarkAngle[size = 4cm,mark = x,arc=ll,color = red](A,O,B)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\end{tikzpicture}
```

14.2.2 mark =|| 选项示例



```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/O,5/0/A,3/4/B}
\tkzMarkAngle[size = 4cm,mark = ||,arc=ll,mkcolor = red](A,O,B)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\end{tikzpicture}
```

14.3 标注角

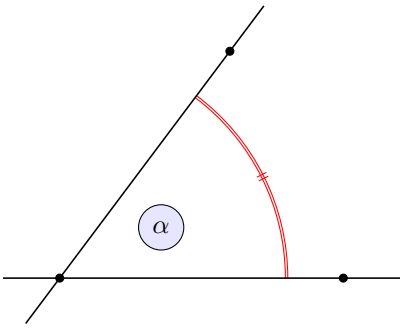
\tkzLabelAngle[选项](A,O,B) 复数形式 \tkzLabelAngles[选项]$(A,O,B)(A',O',B')$

该命令只有一个 dist 选项 (带或不带单位)，该选项可以被 TikZ 的选项 (不带单位) 替代，默认情况下，其单位是 cm

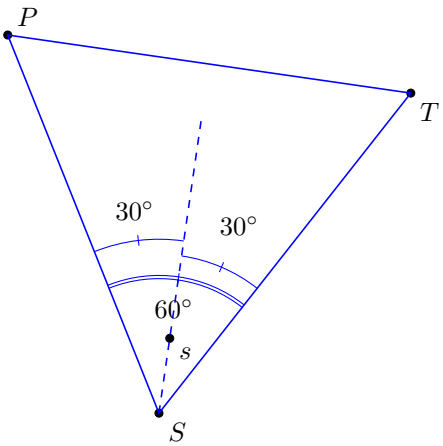
选项	默认值	含义
pos	1	或是 dist，用于控制标注的距离

可以使用 TikZ 的 rotate、shift、below 等选项调整标注的位置

14.3.1 pos 选项示例




```
\begin{tikzpicture}[scale=.75]
\tkzDefPoints{0/0/O,5/0/A,3/4/B}
\tkzMarkAngle[size = 4cm,mark = ||,arc=ll,color = red](A,O,B)
\tkzDrawLines(O,A O,B)
\tkzDrawPoints(O,A,B)
\tkzLabelAngle[pos=2,draw,circle,fill=blue!10](A,O,B){$\alpha$}
\end{tikzpicture}
```




```
\begin{tikzpicture}[rotate=30]
\tkzDefPoint(2,1){S} \tkzDefPoint(7,3){T}
\tkzDefPointBy[rotation=center S angle 60](T) \tkzGetPoint{P}
\tkzDefLine[bisector,normed](T,S,P) \tkzGetPoint{s}
\tkzDrawPoints(S,T,P,s)
\tkzDrawPolygon[color=blue](S,T,P)
\tkzDrawLine[dashed,color=blue,add=0 and 3](S,s)
\tkzLabelPoint[above right](P){$P$}
\tkzLabelPoints(S,T,s)
\tkzMarkAngle[size = 1.8cm,mark = |,arc=ll,color = blue](T,S,P)
\tkzMarkAngle[size = 2.1cm,mark = |,arc=l,color = blue](T,S,s)
\tkzMarkAngle[size = 2.3cm,mark = |,arc=l,color = blue](s,S,P)
\tkzLabelAngle[pos = 1.4](T,S,P){$60^{\circ}$}%
\tkzLabelAngles[pos = 2.7](T,S,s s,S,P){$30^{\circ}$}%
\end{tikzpicture}
```

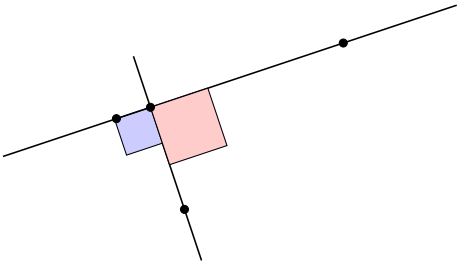
14.4 标记直角

\tkzMarkRightAngle[选项](A,O,B) 复数形式 \tkzMarkRightAngles[选项]$(A,O,B)(A',O',B')$

german 选项用于改变样式，size 选项用于改变尺寸

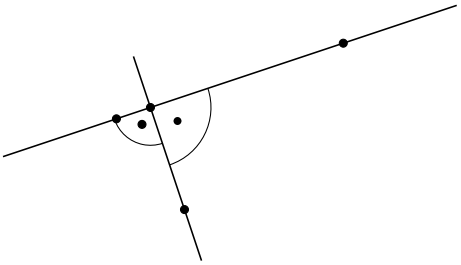
选项	默认值	含义
german	normal	带内点的圆弧
size	0.2	标记边的尺寸

14.4.1 直角标记示例



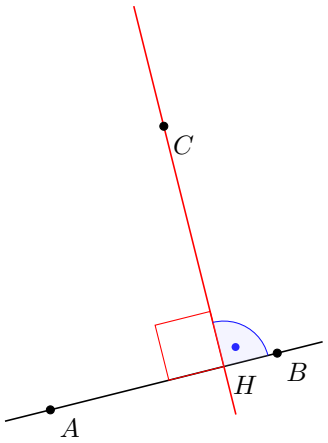
```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,3/1/B,0.9/-1.2/P}
\tkzDefPointBy[projection = onto B-A](P)
\tkzGetPoint{H}
\tkzDrawLines[add=.5 and .5](P,H)
\tkzMarkRightAngle[fill=blue!20,size=.5,draw](A,H,P)
\tkzDrawLines[add=.5 and .5](A,B)
\tkzMarkRightAngle[fill=red!20,size=.8](B,H,P)
\tkzDrawPoints(A,B,P,H)
\end{tikzpicture}
```

14.4.2 使用 german 样式添加直角标记



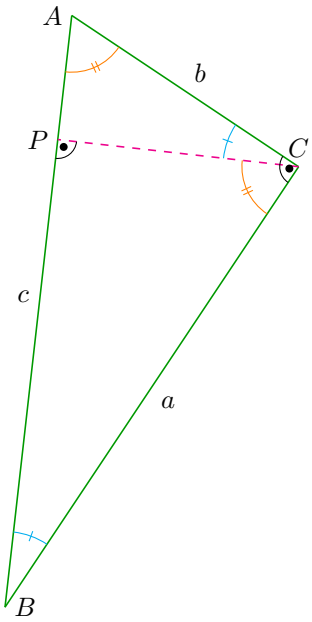
```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,3/1/B,0.9/-1.2/P}
\tkzDefPointBy[projection = onto B-A](P)
\tkzGetPoint{H}
\tkzDrawLines[add=.5 and .5](P,H)
\tkzMarkRightAngle[german,size=.5,draw](A,H,P)
\tkzDrawPoints[](A,B,P,H)
\tkzDrawLines[add=.5 and .5,fill=blue!20](A,B)
\tkzMarkRightAngle[german,size=.8](P,H,B)
\end{tikzpicture}
```

14.4.3 混合样式




```
\begin{tikzpicture}[scale=0.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,1){B}
\tkzDefPoint(2,5){C}
\tkzDefPointBy[projection=onto B-A](C)
\tkzGetPoint{H}
\tkzDrawLine(A,B)
\tkzDrawLine[add = .5 and .2,color=red](C,H)
\tkzMarkRightAngle[,size=1,color=red](C,H,A)
\tkzMarkRightAngle[german,size=.8,color=blue](B,H,C)
\tkzFillAngle[opacity=.2,fill=blue!20,size=.8](B,H,C)
\tkzLabelPoints(A,B,C,H)
\tkzDrawPoints(A,B,C)
\end{tikzpicture}
```

14.4.4 完整示例




```
\begin{tikzpicture}[rotate=-90]
\tkzDefPoint(0,1){A}
\tkzDefPoint(2,4){C}
\tkzDefPointWith[orthogonal normed,K=7](C,A)
\tkzGetPoint{B}
\tkzDrawSegment[green!60!black](A,C)
\tkzDrawSegment[green!60!black](C,B)
\tkzDrawSegment[green!60!black](B,A)
\tkzDrawLine[altitude,dashed,color=magenta](B,C,A)
\tkzGetPoint{P}
\tkzLabelPoint[left](A){$A$}
\tkzLabelPoint[right](B){$B$}
\tkzLabelPoint[above](C){$C$}
\tkzLabelPoint[left](P){$P$}
\tkzLabelSegment[auto](B,A){$c$}
\tkzLabelSegment[auto,swap](B,C){$a$}
\tkzLabelSegment[auto,swap](C,A){$b$}
\tkzMarkAngle[size=1cm,color=cyan,mark=|](C,B,A)
\tkzMarkAngle[size=1cm,color=cyan,mark=|](A,C,P)
\tkzMarkAngle[size=0.75cm,color=orange,mark=||](P,C,B)
\tkzMarkAngle[size=0.75cm,color=orange,mark=||](B,A,C)
\tkzMarkRightAngle[german](A,C,B)
\tkzMarkRightAngle[german](B,P,C)
\end{tikzpicture}
```

14.5 角度测量

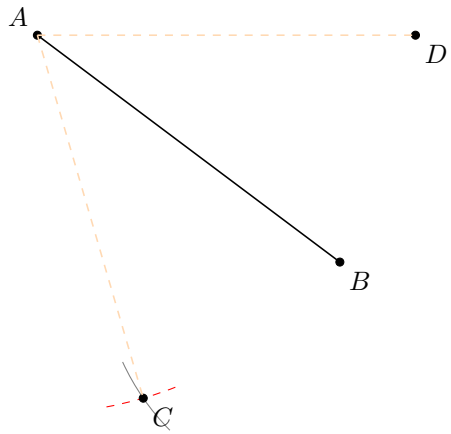
\tkzGetAngle(宏名称)

将角度值以度为单位存入指定宏，可用\tkzAngleResult 命令得到角度值并存入指定宏

参数	样例	含义
宏名称	tkzGetAngle{ang}	\ang 保存了角度值 (度)

14.5.1 tkzGetAngle 命令示例

直线 (AB) 是角 $\widehat{CAD}$ 的角平分线，因此 AD 的斜率为 0，得到 (AB) 的斜率后将其旋转 2 次。




```
\begin{tikzpicture}
\tkzDefPoint(1,5){A} \tkzDefPoint(5,2){B}
\tkzDrawSegment(A,B)
\tkzFindSlopeAngle(A,B) \tkzGetAngle{tkzang}
\tkzDefPointBy[rotation= center A angle \tkzang](B) \tkzGetPoint{C}
\tkzDefPointBy[rotation= center A angle - \tkzang](B) \tkzGetPoint{D}
\tkzCompass[length=1,dashed,color=red](A,C)
\tkzCompass[delta=10,brown](B,C)
\tkzDrawPoints(A,B,C,D)
\tkzLabelPoints(B,C,D)
\tkzLabelPoints[above left](A)
\tkzDrawSegments[style=dashed,color=orange!30](A,C A,D)
\end{tikzpicture}
```

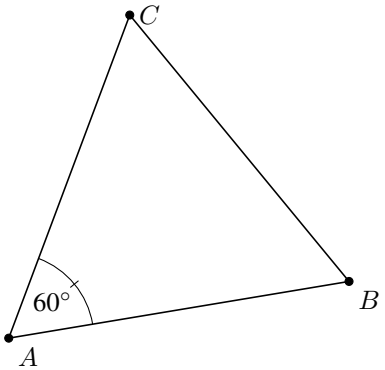
14.6 计算三个点定义的角的角度

\tkzFindAngle$(pt1, pt2, pt3)$

结果保存在\tkzAngleResult 中

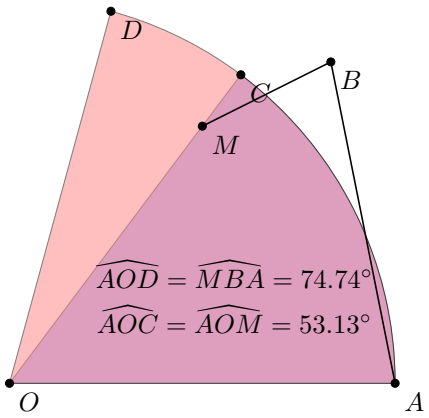
参数	样例	说明
$(pt1, pt2, pt3)$	\tkzFindAngle$(A, B, C)$	\tkzAngleResult 是 $(\overrightarrow{BA}, \overrightarrow{BC})$ 之间的角度
结果在-180 度到 +180 度之间，$pt2$ 是顶点，可以使用\tkzGetAngle 得到结果		

14.6.1 角度测量示例



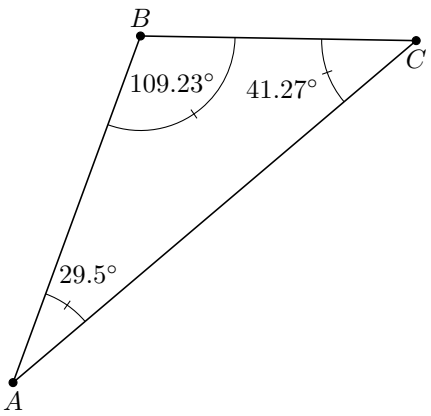

```
\begin{tikzpicture}[scale=.75]
\tkzDefPoint(-1,1){A} \tkzDefPoint(5,2){B}
\tkzDefEquilateral(A,B) \tkzGetPoint{C}
\tkzDrawPolygon(A,B,C)
\tkzFindAngle(B,A,C) \tkzGetAngle{angleBAC}
\edef\angleBAC{\fpeval{round(\angleBAC)}}
\tkzDrawPoints(A,B,C)
\tkzLabelPoints(A,B)
\tkzLabelPoint[right](C){$C$}
\tkzLabelAngle(B,A,C){\angleBAC$^\circ$}
\tkzMarkAngle[size=1.5cm](B,A,C)
\end{tikzpicture}
```

14.6.2 角度计算示例




```
\begin{tikzpicture}[scale=0.85]
\tkzInit[xmin=-1,ymin=-1,xmax=7,ymax=7]
\tkzClip
\tkzDefPoint(0,0){O} \tkzDefPoint (6,0){A}
\tkzDefPoint(5,5){B} \tkzDefPoint (3,4){M}
\tkzFindAngle(A,O,M) \tkzGetAngle{an}
\tkzDefPointBy[rotation=center O angle \an](A)
\tkzGetPoint{C}
\tkzDrawSector[fill = blue!50,opacity=.5](O,A)(C)
\tkzFindAngle(M,B,A)   \tkzGetAngle{am}
\tkzDefPointBy[rotation = center O angle \am](A)
\tkzGetPoint{D}
\tkzDrawSector[fill = red!50,opacity = .5](O,A)(D)
\tkzDrawPoints(O,A,B,M,C,D)
\tkzLabelPoints(O,A,B,M,C,D)
\edef\an{\fpeval{round(\an,2)}}\edef\am{\fpeval{round(\am,2)}}
\tkzDrawSegments(M,B B,A)
\tkzText(3.5,1){$\widehat{AOC}=\widehat{AOM}=\an^{\circ}$}
\tkzText(3.5,1.7){$\widehat{AOD}=\widehat{MBA}=\am^{\circ}$}
\end{tikzpicture}
```

14.6.3 三角形内角计算示例




```
\begin{tikzpicture}[scale=1.25,rotate=30]
\tkzDefPoints{0.5/1.5/A, 3.5/4/B, 6/2.5/C}
\tkzDrawPolygon(A,B,C)
\tkzDrawPoints(A,B,C)
\tkzLabelPoints[below](A,C)
\tkzLabelPoints[above](B)
\tkzMarkAngle[size=1cm](B,C,A)
\tkzFindAngle(B,C,A) \tkzGetAngle{angleBCA}
\edef\angleBCA{\fpeval{round(\angleBCA,2)}}
\tkzLabelAngle[pos = 1.5](B,C,A){$\angleBCA^{\circ}$}
\tkzMarkAngle[size=1cm](C,A,B)
\tkzFindAngle(C,A,B) \tkzGetAngle{angleBAC}
\edef\angleBAC{\fpeval{round(\angleBAC,2)}}
\tkzLabelAngle[pos = 1.4](C,A,B){$\angleBAC^{\circ}$}
\tkzMarkAngle[size=1cm](A,B,C)
\tkzFindAngle(A,B,C) \tkzGetAngle{angleABC}
\edef\angleABC{\fpeval{round(\angleABC,2)}}
\tkzLabelAngle[pos = .6](A,B,C){$\angleABC^{\circ}$}
\end{tikzpicture}
```

14.7 计算斜率

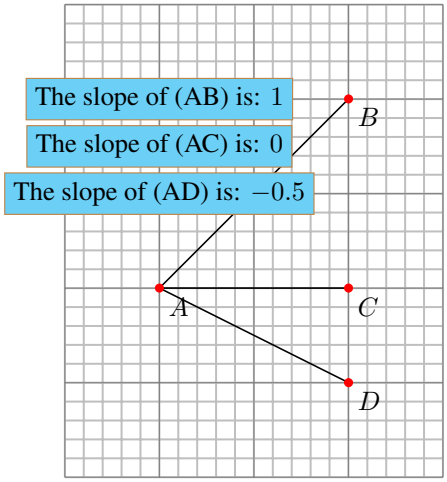
斜率由直线上两个点确定，该命令不检测其存在性

\tkzFindSlope($pt1, pt2$){宏名称}

斜率保存在指定宏中

参数	样例	说明
$(pt1, pt2)$	\tkzFindSlope(A,B){slope}	\slope 通过 $\frac{y_B-y_A}{x_B-x_A}$ 计算

14.7.1 计算斜率示例




```
\begin{tikzpicture}[scale=1.25]
\tkzInit[xmax=4,ymax=5]\tkzGrid[sub]
\tkzDefPoint(1,2){A} \tkzDefPoint(3,4){B}
\tkzDefPoint(3,2){C} \tkzDefPoint(3,1){D}
\tkzDrawSegments(A,B A,C A,D)
\tkzDrawPoints[color=red](A,B,C,D)
\tkzLabelPoints(A,B,C,D)
\tkzFindSlope(A,B){SAB} \tkzFindSlope(A,C){SAC}
\tkzFindSlope(A,D){SAD}
\pgfkeys{/pgf/number format/.cd,fixed,precision=2}
\tkzText[fill=cyan!50,draw=brown](1,4)
{The slope of (AB) is: $\pgfmathprintnumber{\SAB}$}
\tkzText[fill=cyan!50,draw=brown](1,3.5)
{The slope of (AC) is: $\pgfmathprintnumber{\SAC}$}
\tkzText[fill=cyan!50,draw=brown](1,3)
{The slope of (AD) is: $\pgfmathprintnumber{\SAD}$}
\end{tikzpicture}
```

14.8 计算直线与横轴夹角

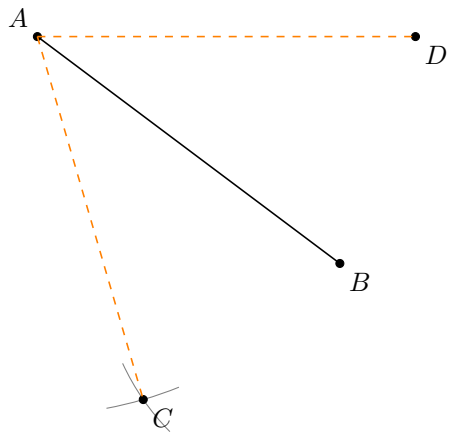
结果在-180 度与 +180 度之间

\tkzFindSlopeAngle($pt1, pt2$)

计算直线 (A, B) 的斜率并保存在\tkzAngleResult 命令中

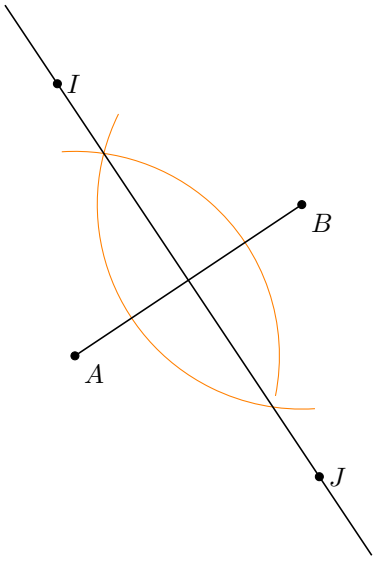
参数	样例	说明
$(pt1, pt2)$	\tkzFindSlopeAngle$(A, B)$	用\tkzGetAngle 保存结果并命名

14.8.1 折叠示例




```
\begin{tikzpicture}
\tkzDefPoint(1,5){A} \tkzDefPoint(5,2){B}
\tkzDrawSegment(A,B)
\tkzFindSlopeAngle(A,B) \tkzGetAngle{tkzang}
\tkzDefPointBy[rotation=center A angle \tkzang](B) \tkzGetPoint{C}
\tkzDefPointBy[rotation=center A angle - \tkzang](B) \tkzGetPoint{D}
\tkzCompass[orange,length=1](A,C)
\tkzCompass[orange,delta=10](B,C)
\tkzDrawPoints(A,B,C,D)
\tkzLabelPoints(B,C,D)
\tkzLabelPoints[above left](A)
\tkzDrawSegments[style=dashed,color=orange](A,C A,D)
\end{tikzpicture}
```

14.8.1.1 中点计算示例




```
\begin{tikzpicture}
\tkzInit
\tkzDefPoint(0,0){A} \tkzDefPoint(3,2){B}
\tkzDefLine[mediator](A,B) \tkzGetPoints{I}{J}
\tkzCalcLength[cm](A,B) \tkzGetLength{dAB}
\tkzFindSlopeAngle(A,B) \tkzGetAngle{tkzangle}
\begin{scope}[rotate=\tkzangle]
\tikzset{arc/.style={color=gray,delta=10}}
\tkzDrawArc[orange,R,arc](B,3/4*\dAB)(120,240)
\tkzDrawArc[orange,R,arc](A,3/4*\dAB)(-45,60)
\tkzDrawLine(I,J)
\tkzDrawSegment(A,B)
\end{scope}
\tkzDrawPoints(A,B,I,J)
\tkzLabelPoints(A,B) \tkzLabelPoints[right](I,J)
\end{tikzpicture}
```

第十五章 扇形

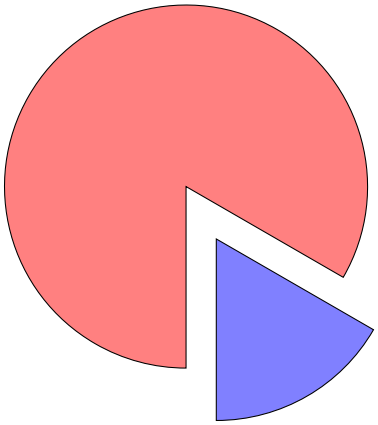
15.1 绘制扇形

\tkzDrawSector[选项]$(O,\dots)(\dots)$

选项	默认值	含义
towards	towards	O 是圆心并且圆弧从 A 到 (OB)
rotate	towards	圆弧从 A 开始并且用角度确定长度
R	towards	给定半径和两个角度
R with nodes	towards	给定半径和两个点
选项	**参数**	**样例**
towards	(pt,pt)(pt)	\tkzDrawSector(O,A)(B)
rotate	(pt,pt)(an)	\tkzDrawSector[rotate,color=red](O,A)(90)
R	(pt,r)(an,an)	\tkzDrawSector[R,color=blue](O,2 cm)(30,90)
R with nodes	(pt,r)(pt,pt)	\tkzDrawSector[R with nodes](O,2 cm)(A,B)
可以使用所有有效的 TikZ 样式		

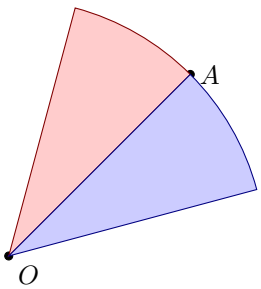
15.1.1 towards 选项示例

towards 是默认选项，同时也可以使用 fill 选项。



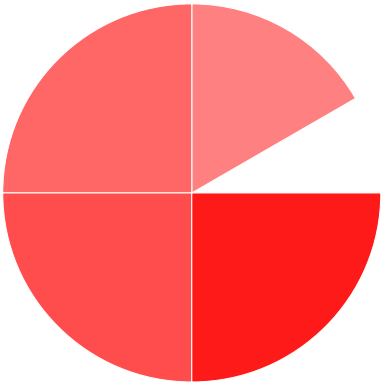
```
\begin{tikzpicture}[scale=0.8]
\tkzDefPoint(0,0){O}
\tkzDefPoint(-30:3){A}
\tkzDefPointBy[rotation = center O angle -60](A)
\tkzDrawSector[fill=red!50](O,A)(tkzPointResult)
\begin{scope}[shift={(-60:1cm)}]
\tkzDefPoint(0,0){O}
\tkzDefPoint(-30:3){A}
\tkzDefPointBy[rotation = center O angle -60](A)
\tkzDrawSector[fill=blue!50](O,tkzPointResult)(A)
\end{scope}
\end{tikzpicture}
```

15.1.2 rotate 选项示例



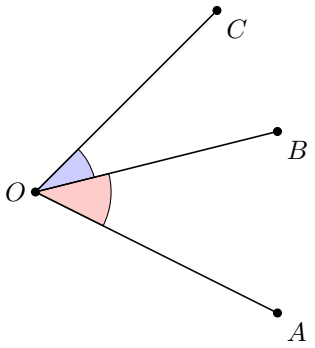

```
\begin{tikzpicture}[scale=1.2]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,2){A}
\tkzDrawPoints(O,A)
\tkzLabelPoints(O)
\tkzLabelPoints[right](A)
\tkzDrawSector[rotate,draw=red!50!black,fill=red!20](O,A)(30)
\tkzDrawSector[rotate,draw=blue!50!black,fill=blue!20](O,A)(-30)
\end{tikzpicture}
```

15.1.3 R 选项示例



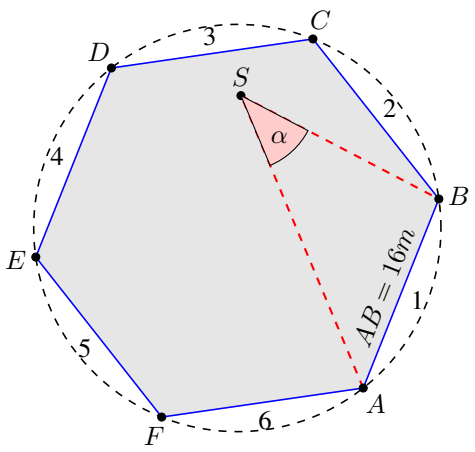
```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){O} \tkzDefPoint(2,-1){A}
\tkzDrawSector[R,draw=white,fill=red!50](O,2cm)(30,90)
\tkzDrawSector[R,draw=white,fill=red!60](O,2cm)(90,180)
\tkzDrawSector[R,draw=white,fill=red!70](O,2cm)(180,270)
\tkzDrawSector[R,draw=white,fill=red!90](O,2cm)(270,360)
\end{tikzpicture}
```

15.1.4 R 选项示例




```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(0,0){O} \tkzDefPoint(4,-2){A}
\tkzDefPoint(4,1){B} \tkzDefPoint(3,3){C}
\tkzDrawSector[R with nodes,fill=blue!20](O,1 cm)(B,C)
\tkzDrawSector[R with nodes,fill=red!20](O,1.25 cm)(A,B)
\tkzDrawSegments(O,A O,B O,C)
\tkzDrawPoints(O,A,B,C)
\tkzLabelPoints(A,B,C)
\tkzLabelPoints[left](O)
\end{tikzpicture}
```

15.1.5 R with nodes 选项示例




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(-1,-2){A} \tkzDefPoint(1,3){B}
\tkzDefRegPolygon[side,sides=6](A,B) \tkzGetPoint{O}
\tkzDrawPolygon[fill=black!10,draw=blue](P1,P...,P6)
\tkzLabelRegPolygon[sep=1.05](O){A,...,F}
\tkzDrawCircle[dashed](O,A)
\tkzLabelSegment[above,sloped,midway](A,B){\(A B = 16m\)}
\foreach \i  [count=\xi from 1]  in {2,...,6,1}
{%
\tkzDefMidPoint(P\xi,P\i)
\path (O) to [pos=1.1] node {\xi} (tkzPointResult) ;
}
\tkzDefRandPointOn[segment = P3-P5] \tkzGetPoint{S}
\tkzDrawSegments[thick,dashed,red](A,S S,B)
\tkzDrawPoints(P1,P...,P6,S)
\tkzLabelPoint[left,above](S){$S$}
\tkzDrawSector[R with nodes,fill=red!20](S,2 cm)(A,B)
\tkzLabelAngle[pos=1.5](A,S,B){$\alpha$}
\end{tikzpicture}
```

15.2 填充扇形

\tkzFillSector[选项]$(O,\dots)(\dots)$

选项	默认值	含义
towards	towards	O 是圆心并且圆弧从 A 到 (OB)
rotate	towards	圆弧从 A 开始并且用角度确定长度
R	towards	给定半径和两个角度
R with nodes	towards	给定半径和两个点
选项	**参数**	**样例**
towards	(pt,pt)(pt)	\tkzFillSector(O,A)(B)
rotate	(pt,pt)(an)	\tkzFillSector[rotate,color=red](O,A)(90)
R	(pt,r)(an,an)	\tkzFillSector[R,color=blue](O,2 cm)(30,90)
R with nodes	(pt,r)(pt,pt)	\tkzFillSector[R with nodes](O,2 cm)(A,B)
可以使用所有有效的 TikZ 样式		

15.2.1 towards 选项示例

towards 是默认选项，该命令不绘制轮廓，仅对区域进行着色。

```
\begin{tikzpicture}[scale=.6]
\tkzDefPoint(0,0){O}
\tkzDefPoint(-30:3){A}
\tkzDefPointBy[rotation = center O angle -60](A)
\tkzFillSector[fill=red!50](O,A)(tkzPointResult)
\begin{scope}[shift={(-60:1cm)}]
\tkzDefPoint(0,0){O}
\tkzDefPoint(-30:3){A}
\tkzDefPointBy[rotation = center O angle -60](A)
\tkzFillSector[color=blue!50](O,tkzPointResult)(A)
\end{scope}
\end{tikzpicture}
```

15.2.2 rotate 选项示例

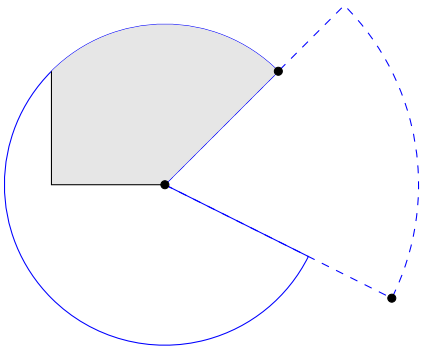

```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){O} \tkzDefPoint(2,2){A}
\tkzFillSector[rotate,color=red!20](O,A)(30)
\tkzFillSector[rotate,color=blue!20](O,A)(-30)
\end{tikzpicture}
```

15.3 扇形裁剪

\tkzClipSector[选项]$(O,\dots)(\dots)$

选项	默认值	含义
towards	towards	O 是圆心并且圆弧从 A 到 (OB)
rotate	towards	圆弧从 A 开始并且用角度确定长度
R	towards	给定半径和两个角度
选项	**参数**	**样例**
towards	(pt,pt)(pt)	\tkzClipSector(O,A)(B)
rotate	(pt,pt)(an)	\tkzClipSector[rotate,color=red](O,A)(90)
R	(pt,r)(an,an)	\tkzClipSector[R,color=blue](O,2 cm)(30,90)
可以使用所有有效的 TikZ 样式		

15.3.1 示例



```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,-1){A}
\tkzDefPoint(1,1){B}
\tkzDrawSector[color=blue,dashed](O,A)(B)
\tkzDrawSector[color=blue](O,B)(A)
\tkzClipBB
\begin{scope}
\tkzClipSector(O,B)(A)
\draw[fill=gray!20] (-1,0) rectangle (3,3);
\end{scope}
\tkzDrawPoints(A,B,O)
\end{tikzpicture}
```

第十六章 圆弧

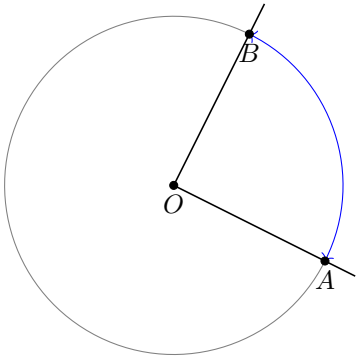
16.1 绘制圆弧

\tkzDrawArc[选项]($O,\dots$)($\dots$)

选项	默认值	含义
towards	towards	O 是圆心并且圆弧从 A 到 (OB)
rotate	towards	圆弧从 A 开始并且用角度确定长度
R	towards	给定半径和两个角度
R with nodes	towards	给定半径和两个点
angles	towards	给定半径和两个点
delta	0	角度加上两个边
选项	**参数**	**样例**
towards	(pt,pt)(pt)	\tkzDrawArc[delta=10](O,A)(B)
rotate	(pt,pt)(an)	\tkzDrawArc[rotate,color=red](O,A)(90)
R	(pt,r)(an,an)	\tkzDrawArc[R](O,2 cm)(30,90)
R with nodes	(pt,r)(pt,pt)	\tkzDrawArc[R with nodes](O,2 cm)(A,B)
angles	(pt,pt)(an,an)	\tkzDrawArc[angles](O,A)(0,90)
可以使用所有有效的 TikZ 样式		

16.1.1 towards 选项示例 1

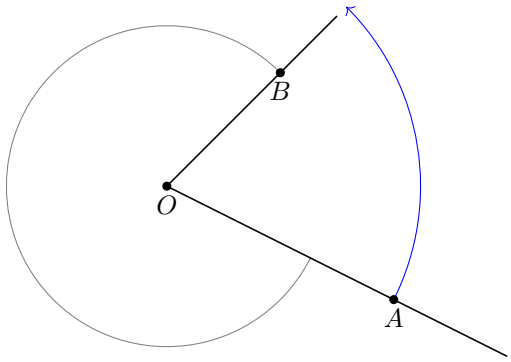
towards 是默认选项，第 1 个例子里的圆弧从 A 开始到 B 结束。当然，从 B 到 A 会得到不同的结果。圆弧凸向由逆时针方向确定。




```
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,-1){A}
\tkzDefPointBy[rotation= center O angle 90](A)
\tkzGetPoint{B}
\tkzDrawArc[color=blue,<->](O,A)(B)
\tkzDrawArc(O,B)(A)
\tkzDrawLines[add = 0 and .2](O,A O,B)
\tkzDrawPoints(O,A,B)
\tkzLabelPoints[below](O,A,B)
\end{tikzpicture}
```

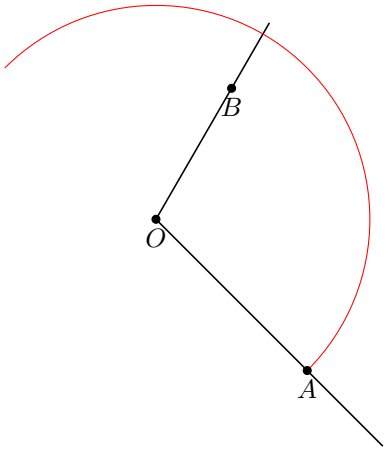
16.1.2 towards 选项示例 2

该例中，圆弧从 A 开始，直到 (OB) 结束。



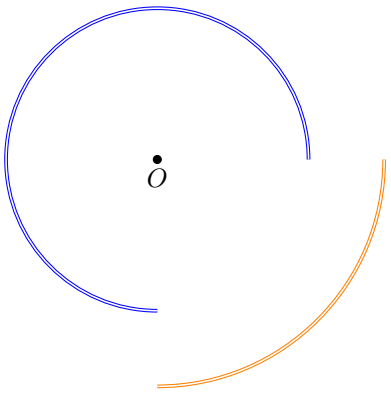

```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,-1){A}
\tkzDefPoint(1,1){B}
\tkzDrawArc[color=blue,->](O,A)(B)
\tkzDrawArc[color=gray](O,B)(A)
\tkzDrawLines[add = 0 and .5](O,A O,B)
\tkzDrawPoints(O,A,B)
\tkzLabelPoints[below](O,A,B)
\end{tikzpicture}
```

16.1.3 rotate 选项示例



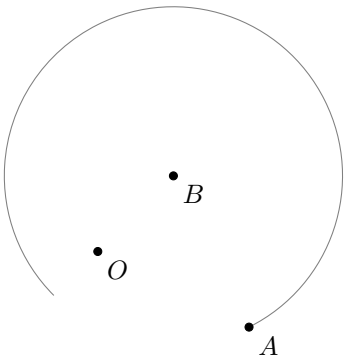

```
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,-2){A}
\tkzDefPoint(60:2){B}
\tkzDrawLines[add = 0 and .5](O,A O,B)
\tkzDrawArc[rotate,color=red](O,A)(180)
\tkzDrawPoints(O,A,B)
\tkzLabelPoints[below](O,A,B)
\end{tikzpicture}
```

16.1.4 R 选项示例




```
\begin{tikzpicture}
\tkzDefPoints{0/0/O}
%\tikzset{compass style/.append style={<->}}
\tkzDrawArc[R,color=orange,double](O,3cm)(270,360)
\tkzDrawArc[R,color=blue,double](O,2cm)(0,270)
\tkzDrawPoint(O)
\tkzLabelPoint[below](O){$O$}
\end{tikzpicture}
```

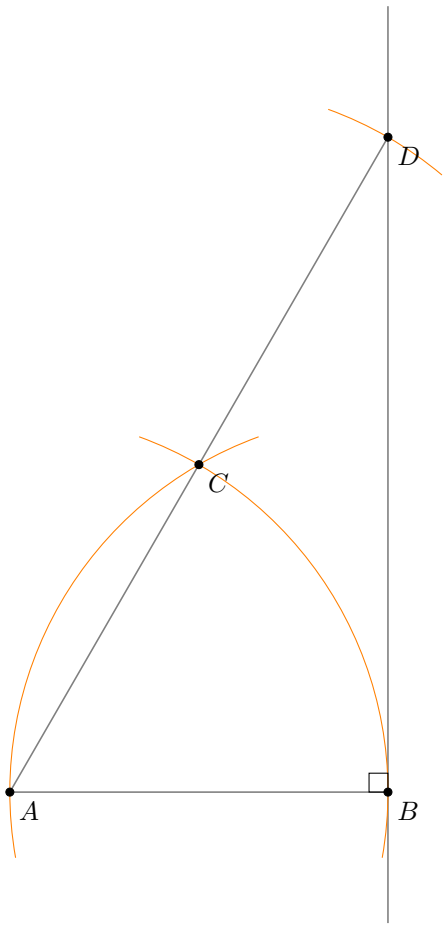
16.1.5 R with nodes 选项示例




```
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(2,-1){A}
\tkzDefPoint(1,1){B}
\tkzCalcLength(B,A)\tkzGetLength{radius}
\tkzDrawArc[R with nodes](B,\radius pt)(A,O)
\tkzDrawPoints(O,A,B)
\tkzLabelPoints(O,A,B)
\end{tikzpicture}
```

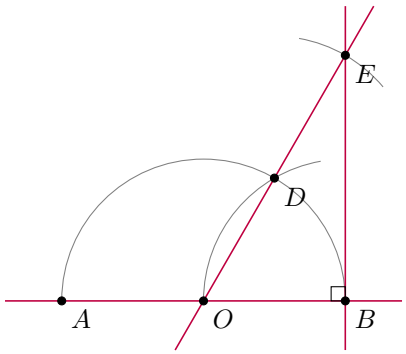
16.1.6 delta 选项示例

该选项与\tkzCompass 结果类似，它能够延伸圆弧，`delta` 的单位是度。



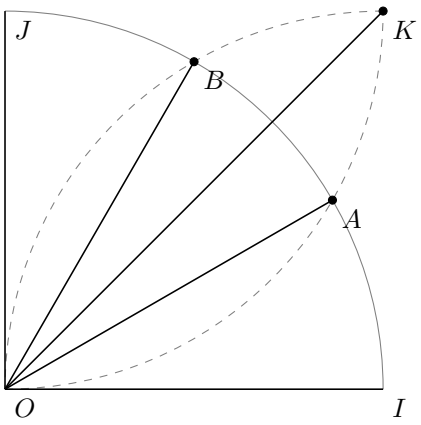

```
\begin{tikzpicture}
\tkzDefPoint(0,0){A}
\tkzDefPoint(5,0){B}
\tkzDefPointBy[rotation= center A angle 60](B)
\tkzGetPoint{C}
\tkzSetUpLine[color=gray]
\tkzDefPointBy[symmetry= center C](A)
\tkzGetPoint{D}
\tkzDrawSegments(A,B A,D)
\tkzDrawLine(B,D)
\tkzSetUpCompass[color=orange]
\tkzDrawArc[orange,delta=10](A,B)(C)
\tkzDrawArc[orange,delta=10](B,C)(A)
\tkzDrawArc[orange,delta=10](C,D)(D)
\tkzDrawPoints(A,B,C,D)
\tkzLabelPoints(A,B,C,D)
\tkzMarkRightAngle(D,B,A)
\end{tikzpicture}
```

16.1.7 angles 选项示例 1




```
\begin{tikzpicture}[scale=.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(5,0){B}
\tkzDefPoint(2.5,0){O}
\tkzDefPointBy[rotation=center O angle 60](B)
\tkzGetPoint{D}
\tkzDefPointBy[symmetry=center D](O)
\tkzGetPoint{E}
\tkzSetUpLine[color=purple]
\tkzDrawArc[angles](O,B)(0,180)
\tkzDrawArc[angles,](B,O)(100,180)
\tkzCompass[delta=20](D,E)
\tkzDrawLines(A,B O,E B,E)
\tkzDrawPoints(A,B,O,D,E)
\tkzLabelPoints(A,B,O,D,E)
\tkzMarkRightAngle(O,B,E)
\end{tikzpicture}
```

16.1.8 angles 选项示例 2




```
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(5,0){I}
\tkzDefPoint(0,5){J}
\tkzInterCC(O,I)(I,O)\tkzGetPoints{B}{C}
\tkzInterCC(O,I)(J,O)\tkzGetPoints{D}{A}
\tkzInterCC(I,O)(J,O)\tkzGetPoints{L}{K}
\tkzDrawArc[angles](O,I)(0,90)
\tkzDrawArc[angles,color=gray,style=dashed](I,O)(90,180)
\tkzDrawArc[angles,color=gray,style=dashed](J,O)(-90,0)
\tkzDrawPoints(A,B,K)
\foreach \point in {I,A,B,J,K}{\tkzDrawSegment(O,\point)}
\tkzLabelPoints(A,B,I,J,K,O)
\end{tikzpicture}
```

第十七章 杂项命令

17.1 复制线段

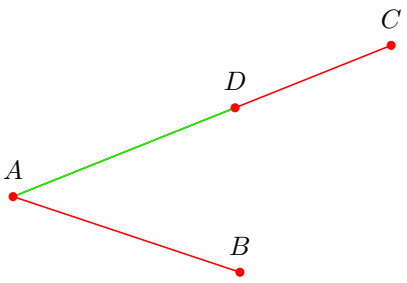
在给定的射线上复制与给定线段长度相同的线段

\tkzDuplicateSegment($pt1, pt2$)($pt3, pt4$)

该命令实际是一个点的定义命令

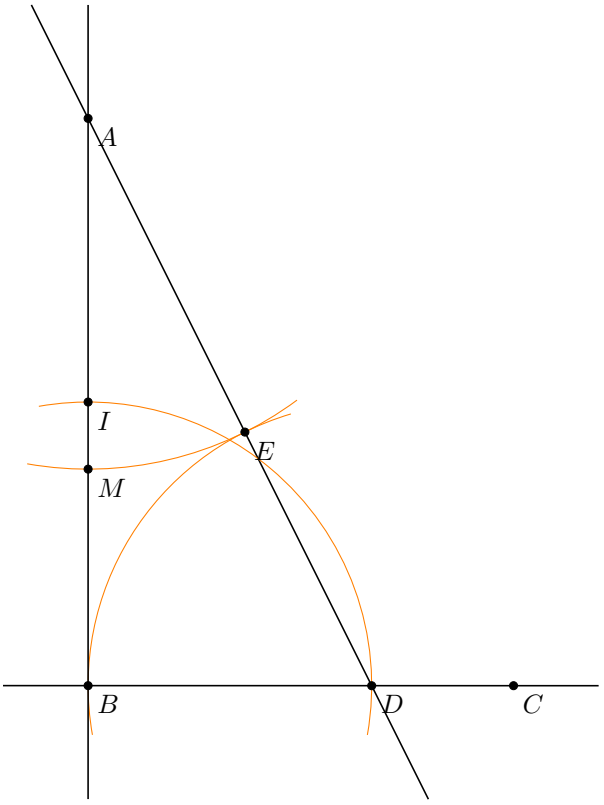
参数	样例	说明
($pt1, pt2$)($pt3, pt4$)	\tkzDuplicateSegment(A,B)(E,F)	$AB = EC$

17.1.1 示例 1




```
\begin{tikzpicture}
\tkzDefPoints{0/0/A,3/-1/B,5/2/C}
\tkzDrawSegments[red](A,B A,C)
\tkzDuplicateSegment(A,B)(A,C) \tkzGetPoint{D}
\tkzDrawSegment[green](A,D)
\tkzDrawPoints[color=red](A,B,C,D)
\tkzLabelPoints[above=3pt](A,B,C,D)
\end{tikzpicture}
```

17.1.2 黄金分割示例




```
\begin{tikzpicture}[rotate=-90,scale=.75]
\tkzDefPoint(0,0){A}
\tkzDefPoint(10,0){B}
\tkzDefMidPoint(A,B)
\tkzGetPoint{I}
\tkzDefPointWith[orthogonal,K=-.75](B,A)
\tkzGetPoint{C}
\tkzInterLC(B,C)(B,I)  \tkzGetSecondPoint{D}
\tkzDuplicateSegment(B,D)(D,A) \tkzGetPoint{E}
\tkzInterLC(A,B)(A,E)   \tkzGetPoints{N}{M}
\tkzDrawArc[orange,delta=10](D,E)(B)
\tkzDrawArc[orange,delta=10](A,M)(E)
\tkzDrawLines(A,B B,C A,D)
\tkzDrawArc[orange,delta=10](B,D)(I)
\tkzDrawPoints(A,B,D,C,M,I,E)
\tkzLabelPoints(A,B,D,C,M,I,E)
\end{tikzpicture}
```

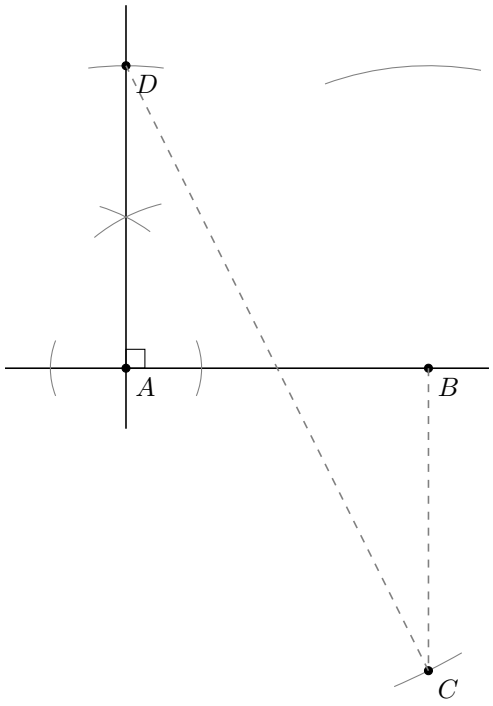
17.2 计算线段长度

也可以用 TikZ 的 veclen 计算长度，该选项能够计算 A 点和 B 点间的距离 AB。但 TikZ 计算精度不足，因此该命令用 xfp 宏包实现计算，虽然其计算慢，但精度高

\tkzCalcLength[选项]($pt1, pt2$){宏名称}

参数	样例	说明
($pt1, pt2$){宏名称}	\tkzCalcLength(A, B){dAB}	\dAB 得到 AB 的长度，单位是 pt
选项	**样例**	**说明**
cm	\tkzCalcLength[cm](A,B){dAB}	\dAB 得到 AB 的长度，单位是 cm

17.2.1 绘制尺规标记




```
\begin{tikzpicture}[scale=1]
\tkzDefPoint(0,0){A}
\tkzDefPoint(4,0){B}
\tkzDrawLine[add= .4 and .2](A,B)
\tkzCalcLength[cm](A,B)\tkzGetLength{dAB}
\tkzDefLine[perpendicular=through A](A,B)
\tkzDrawLine(A,tkzPointResult)
\tkzGetPoint{D}
\tkzShowLine[orthogonal=through A,gap=2](A,B)
\tkzMarkRightAngle(B,A,D)
\tkzVecKOrth[-1](B,A)\tkzGetPoint{C}
%VecK 矢量，Orth 正交
\tkzCompasss(A,D D,C)
\tkzDrawArc[R](B,\dAB)(80,110)
\tkzDrawPoints(A,B,C,D)
\tkzDrawSegments[color=gray,style=dashed](B,C C,D)
\tkzLabelPoints(A,B,C,D)
\end{tikzpicture}
```

17.3 pt,cm 转换

\tkzpttocm(nombre){宏名称}

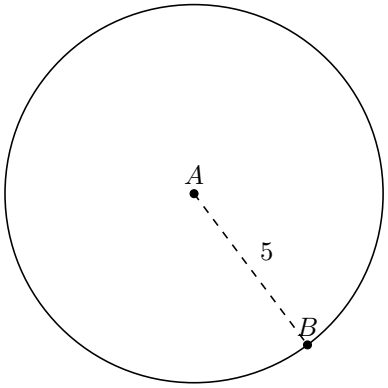
参数	样例	说明
(number){宏名称}	\tkzpttocm(120){len}	\len 得到 cm 值

\tkzcmtopt(nombre){宏名称}

参数	样例	说明
(number){宏名称}	\tkzpttocm(5){len}	\len 得到 pt 值

17.3.1 示例代码

\tkzDefCircle[radius](A,B) 命令定义一个圆，可以用\tkzGetLength 命令得到半径，但其单位是 pt。



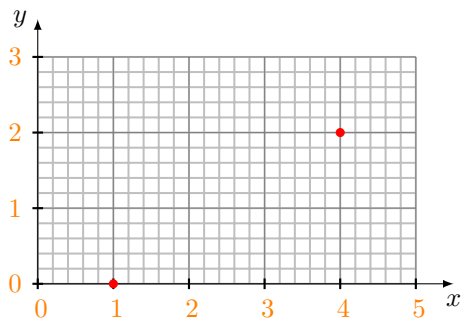

```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,-4){B}
\tkzDefCircle[through](A,B)
\tkzGetLength{rABpt}
\tkzpttocm(\rABpt){rABcm}
\tkzDrawCircle(A,B)
\tkzDrawPoints(A,B)
\tkzLabelPoints[above](A,B)
\tkzDrawSegment[dashed](A,B)
\tkzLabelSegment(A,B){$\pgfmathprintnumber{\rABcm}$}
\end{tikzpicture}
```

17.4 提取点的坐标分量

\tkzGetPointCoord(*A*){宏名称}

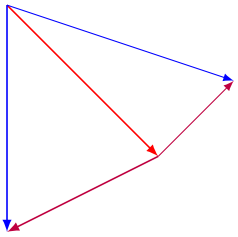
参数	样例	说明
(pt){宏名称}	\tkzGetPointCoord(A){A}	\Ax 和\Ay 保存点 *A* 的坐标分量
将点的坐标分量保存在两个宏中，如果宏名称是 p, 则将坐标分量保存在\px 和\py 宏中，单位是 cm		

17.4.1 点定义示例




```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=3]
\tkzGrid[sub,orange]     \tkzAxeXY%Axe-轴线
\tkzDefPoint(1,0){A}     \tkzDefPoint(4,2){B}
\tkzGetPointCoord(A){a} \tkzGetPointCoord(B){b}
\tkzDefPoint(\ax,\ay){C}\tkzDefPoint(\bx,\by){D}
\tkzDrawPoints[color=red](C,D)
\end{tikzpicture}
```

17.4.2 求向量和示例



```
\begin{tikzpicture}[>=latex]
\tkzDefPoint(1,4){a} \tkzDefPoint(3,2){b} \tkzDefPoint(1,1){c}
\tkzDrawSegment[->,red](a,b) \tkzGetPointCoord(c){c}
\draw[color=blue,->](a) - ([shift=(b)]\cx,\cy) ;
\draw[color=purple,->](b) - ([shift=(b)]\cx,\cy) ;
\tkzDrawSegment[->,blue](a,c)
\tkzDrawSegment[->,purple](b,c)
\end{tikzpicture}
```

第十八章 尺规标记

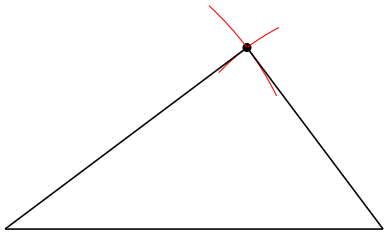
18.1 绘制尺规标记

\tkzCompass[选项](A,B) 复数形式 \tkzCompasss[选项]$(pt1,pt2 \quad pt3,pt4 \quad \ldots)$

该命令绘制尺规标记，即一小段圆弧。使用该命令时，须指定圆心。
可以使用 TikZ 的 style、color、line thickness 等样式设置标记外观。可以使用 length 或 delta 选项指定标记长度

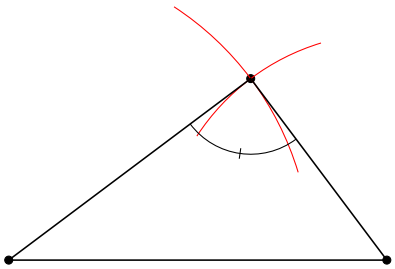
选项	默认值	含义
delta	0 (deg)	延伸长度 (度)
length	1 (cm)	圆弧长度 (cm)

18.1.1 length 选项示例



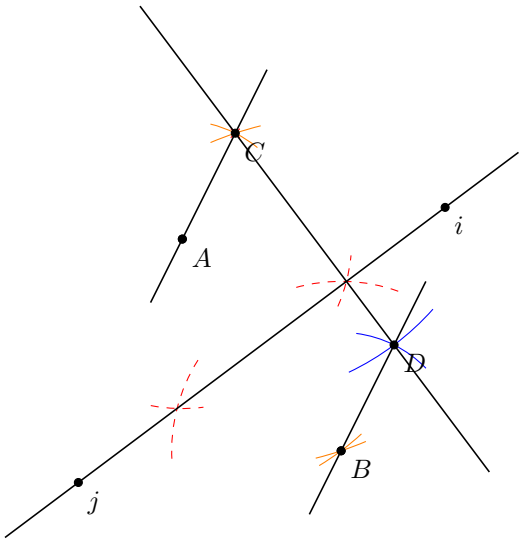
```
\begin{tikzpicture}
\tkzDefPoint(1,1){A} \tkzDefPoint(6,1){B}
\tkzInterCC[R](A,4cm)(B,3cm) \tkzGetPoints{C}{D}
\tkzDrawPoint(C)
\tkzCompass[color=red,length=1.5](A,C)
\tkzCompass[color=red](B,C)
\tkzDrawSegments(A,B A,C B,C)
\end{tikzpicture}
```

18.1.1.1 delta 选项示例



```
\begin{tikzpicture}
\tkzDefPoint(0,0){A} \tkzDefPoint(5,0){B}
\tkzInterCC[R](A,4cm)(B,3cm) \tkzGetPoints{C}{D}
\tkzDrawPoints(A,B,C)
\tkzCompass[color=red,delta=20](A,C)
\tkzCompass[color=red,delta=20](B,C)
\tkzDrawPolygon(A,B,C) \tkzMarkAngle(A,C,B)
\end{tikzpicture}
```

18.1.2 绘制多个尺规标记示例



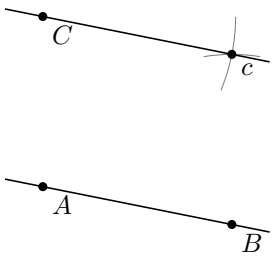

```
\begin{tikzpicture}[scale=.7]
\tkzDefPoint(2,2){A} \tkzDefPoint(5,-2){B} \tkzDefPoint(3,4){C}
\tkzDrawPoints(A,B)
\tkzDrawPoint[color=red,shape=cross out](C)
\tkzCompasss[color=orange](A,B A,C B,C C,B)
\tkzShowLine[mediator,color=red,dashed,length = 2](A,B)
\tkzShowLine[parallel = through C,color=blue,length=2](A,B)
\tkzDefLine[mediator](A,B) \tkzGetPoints{i}{j}
\tkzDefLine[parallel=through C](A,B) \tkzGetPoint{D}
\tkzDrawLines[add=.6 and .6](C,D A,C B,D)
\tkzDrawLines(i,j) \tkzDrawPoints(A,B,C,i,j,D)
\tkzLabelPoints(A,B,C,i,j,D)
\end{tikzpicture}
```

18.2 显示直线尺规标记

\tkzShowLine[选项](*pt*1, *pt*2) 或 (*pt*1, *pt*2, *pt*3)

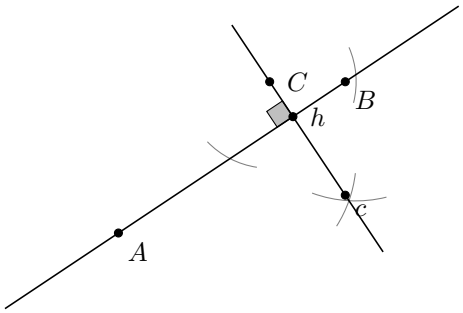
选项	默认值	含义
mediator	mediator	中垂线
perpendicular	mediator	垂线
orthogonal	mediator	同上
parallel	mediator	平行线
bisector	mediator	角平分线
K	1	三角形内圆
length	1	圆弧长度，单位是 cm
ratio	.5	圆弧长度比例
gap	2	符号间隙
size	1	圆弧半径 (参见 bisector)

18.2.1 parallel 选项示例



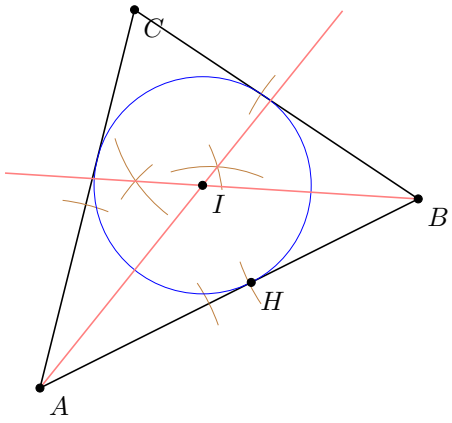

```
\begin{tikzpicture}
\tkzDefPoints{-1.5/-0.25/A,1/-0.75/B,-1.5/2/C}
\tkzDrawLine(A,B)
\tkzDefLine[parallel=through C](A,B) \tkzGetPoint{c}
\tkzShowLine[parallel=through C](A,B)
\tkzDrawLine(C,c) \tkzDrawPoints(A,B,C,c)
\tkzLabelPoints(A,B,C,c)
\end{tikzpicture}
```

18.2.2 perpendicular 选项示例



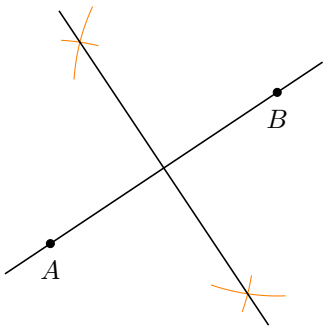

```
\begin{tikzpicture}
\tkzDefPoints{0/0/A, 3/2/B, 2/2/C}
\tkzDefLine[perpendicular=through C,K=-.5](A,B) \tkzGetPoint{c}
\tkzShowLine[perpendicular=through C,K=-.5,gap=3](A,B)
\tkzDefPointBy[projection=onto A-B](c) \tkzGetPoint{h}
\tkzMarkRightAngle[fill=lightgray](A,h,C)
\tkzDrawLines[add=.5 and .5](A,B C,c)
\tkzDrawPoints(A,B,C,h,c)
\tkzLabelPoints(A,B,c) \tkzLabelPoints[right=3pt](C,h)
\end{tikzpicture}
```

18.2.3 bisector 选项示例




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoints{0/0/A, 4/2/B, 1/4/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpCompass[color=brown,line width=.1 pt]
\tkzDefLine[bisector](B,A,C)  \tkzGetPoint{a}
\tkzDefLine[bisector](C,B,A)  \tkzGetPoint{b}
\tkzInterLL(A,a)(B,b) \tkzGetPoint{I}
\tkzDefPointBy[projection = onto A-B](I) \tkzGetPoint{H}
\tkzShowLine[bisector,size=2,gap=3,blue](B,A,C)
\tkzShowLine[bisector,size=2,gap=3,blue](C,B,A)
\tkzDrawCircle[radius,color=blue,line width=.2pt](I,H)
\tkzDrawSegments[color=red!50](I,tkzPointResult)
\tkzDrawLines[add=0 and -0.3,color=red!50](A,a B,b)
\tkzDrawPoints(A,B,C,I,H) \tkzLabelPoints(A,B,C,I,H)
\end{tikzpicture}
```

18.2.4 mediator 选项示例



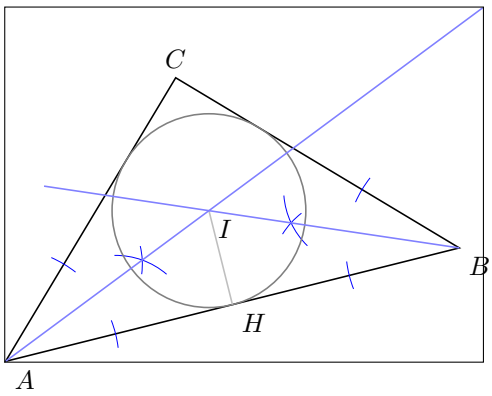

```
\begin{tikzpicture}
\tkzDefPoint(2,2){A}
\tkzDefPoint(5,4){B}
\tkzDrawPoints(A,B)
\tkzShowLine[mediator,color=orange,length=1](A,B)
\tkzGetPoints{i}{j}
\tkzDrawLines[add=-0.1 and -0.1](i,j)
\tkzDrawLines(A,B)
\tkzLabelPoints[below =3pt](A,B)
\end{tikzpicture}
```

18.3 设置尺规标记样式

\tkzSetUpCompass[选项]

选项	默认值	含义
line width	0.4pt	线宽
color	black!50	颜色
style	solid	线型：solid、dashed、dotted、...

18.3.1 示例



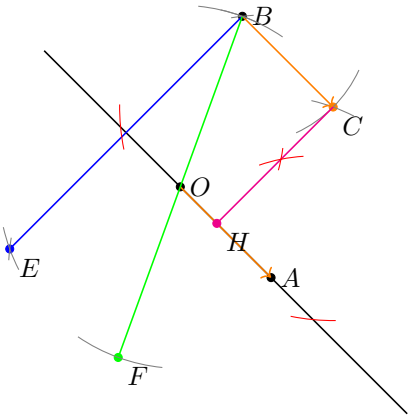

```
\begin{tikzpicture}[scale=.75,
showbi/.style={bisector,size=2,gap=3}]
\tkzSetUpCompass[color=blue,line width=.3 pt]
\tkzDefPoints{0/1/A, 8/3/B, 3/6/C}
\tkzDrawPolygon(A,B,C)
\tkzDefLine[bisector](B,A,C) \tkzGetPoint{a}
\tkzDefLine[bisector](C,B,A) \tkzGetPoint{b}
\tkzShowLine[showbi](B,A,C)
\tkzShowLine[showbi](C,B,A)
\tkzInterLL(A,a)(B,b) \tkzGetPoint{I}
\tkzDefPointBy[projection= onto A-B](I)
\tkzGetPoint{H}
\tkzDrawCircle[radius,color=gray](I,H)
\tkzDrawSegments[color=gray!50](I,H)
\tkzDrawLines[add=0 and -.2,color=blue!50](A,a B,b)
\tkzShowBB
\tkzLabelPoints(A,B,I,H)
\tkzLabelPoints[above](C)
\end{tikzpicture}
```

18.4 显示部分变换过程尺规标记

\tkzShowTransformation[选项]($pt1, pt2$) 或 ($pt1, pt2, pt3$)

选项	默认值	含义
reflection= over pt1–pt2	reflection	正交对称
symmetry=center pt	reflection	中心对称
projection=onto pt1–pt2	reflection	投影
translation=from pt1 to pt2	reflection	平移
K	1	三角形内的圆
length	1	圆弧长度
ratio	.5	圆弧长度比例
gap	2	标记间隙
size	1	圆弧半径 (参见 bisector)

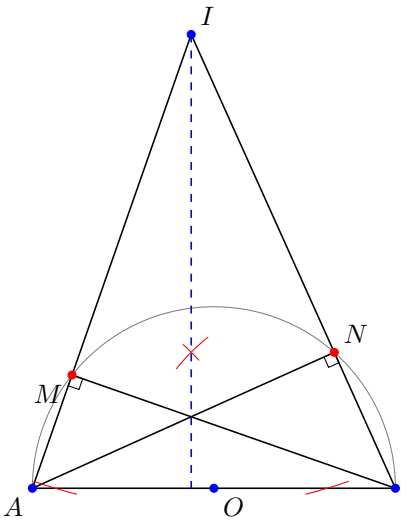
18.4.1 示例 1




```
\begin{tikzpicture}[scale=.6]
\tkzDefPoint(0,0){O} \tkzDefPoint(2,-2){A} \tkzDefPoint(70:4){B}
\tkzDrawPoints(A,O,B) \tkzLabelPoints[right](A,O,B)
\tkzDrawLine[add= 1.5 and 1.5](O,A)
\tkzDefPointBy[translation=from O to A](B) \tkzGetPoint{C}
\tkzDrawPoint[color=orange](C)  \tkzLabelPoints(C)
\tkzShowTransformation[translation=from O to A,length=2](B)
\tkzDrawSegments[->,color=orange](O,A B,C)
\tkzDefPointBy[reflection=over O-A](B) \tkzGetPoint{E}
\tkzDrawSegment[blue](B,E)
\tkzDrawPoint[color=blue](E)\tkzLabelPoints(E)
\tkzShowTransformation[reflection=over O-A,size=2](B)
\tkzDefPointBy[symmetry=center O](B) \tkzGetPoint{F}
\tkzDrawSegment[color=green](B,F)
\tkzDrawPoint[color=green](F)\tkzLabelPoints(F)
\tkzShowTransformation[symmetry=center O,length=2](B)
\tkzDefPointBy[projection=onto O-A](C) \tkzGetPoint{H}
\tkzDrawSegments[color=magenta](C,H)
\tkzDrawPoint[color=magenta](H)\tkzLabelPoints(H)
\tkzShowTransformation[projection=onto O-A,color=red,size=3,gap=-2](C)
\end{tikzpicture}
```

18.4.1.1 示例 2

后续会再次使用这个示例，但不绘制尺规标记。



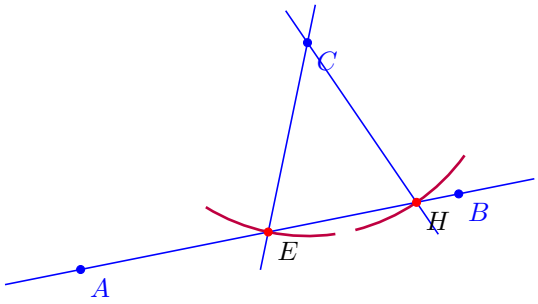

```
\begin{tikzpicture}[scale=.6]
\tkzDefPoints{0/0/A,8/0/B,3.5/10/I}
\tkzDefMidPoint(A,B) \tkzGetPoint{O}
\tkzDefPointBy[projection=onto A-B](I)
\tkzGetPoint{J}
\tkzInterLC(I,A)(O,A)  \tkzGetPoints{M'}{M}
\tkzInterLC(I,B)(O,A)  \tkzGetPoints{N}{N'}
\tkzDrawSemiCircle[diameter](A,B)
\tkzDrawSegments(I,A I,B A,B B,M A,N)
\tkzMarkRightAngles(A,M,B A,N,B)
\tkzDrawSegment[style=dashed,color=blue](I,J)
\tkzShowTransformation[projection=onto A-B,color=red,size=3,gap=-3](I)
\tkzDrawPoints[color=red](M,N)
\tkzDrawPoints[color=blue](O,A,B,I)
\tkzLabelPoints(O)
\tkzLabelPoints[above right](N,I)
\tkzLabelPoints[below left](M,A)
\end{tikzpicture}
```

18.5 差分点

\tkzDefEquiPoints[选项](*pt*1, *pt*2)

参数	默认值	含义
(*pt*1, *pt*2)	默认值	用于定义直线的两个点
选项	**默认值**	**含义**
dist	2 cm	直线上两点间距离的一半
from=pt	无	参考点
show	false	如为 true，则显示尺规标记
/compass/delta	0	尺规标记尺寸

18.5.1 示例




```
\begin{tikzpicture}
\tkzSetUpCompass[color=purple,line width=1pt]
\tkzDefPoint(0,1){A} \tkzDefPoint(5,2){B} \tkzDefPoint(3,4){C}
\tkzDefEquiPoints[from=C,dist=1,show,/tkzcompass/delta=20](A,B)
\tkzGetPoints{E}{H}
\tkzDrawLines[color=blue](C,E C,H A,B)
\tkzDrawPoints[color=blue](A,B,C)
\tkzDrawPoints[color=red](E,H)
\tkzLabelPoints(E,H)
\tkzLabelPoints[color=blue](A,B,C)
\end{tikzpicture}
```

第十九章 量角器

基于 *Yves Combe* 的方法，其工作原理更为简单，仅半条直线 (射线)，量角器原点位于点 O，射线方向由 A 确定。角度方向由指定的测量圆方向决定。

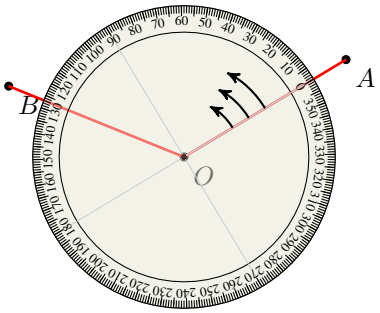
19.1 绘制量角器

\tkzProtractor[选项](O, A)

选项	默认值	含义
lw	0.4 pt	线宽
scale	1	比例: 用于调整量角器尺寸
return	false	反向测量圆

19.1.1 正向圆量角器

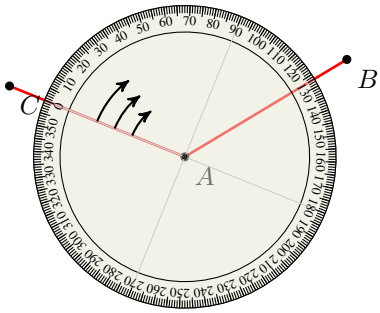
正向测量圆方向




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(0,0){O}
\tkzDefShiftPoint[O](31:5){A}
\tkzDefShiftPoint[O](158:5){B}
\tkzDrawPoints(A,B,O) \tkzLabelPoints(A,B,O)
\tkzDrawSegments[color = red,line width = 1pt](O,A O,B)
\tkzProtractor[scale = 1](O,A)
\end{tikzpicture}
```

19.1.2 反向圆量角器

逆向测量圆方向




```
\begin{tikzpicture}[scale=.5]
\tkzDefPoint(2,3){A}
\tkzDefShiftPoint[A](31:5){B}
\tkzDefShiftPoint[A](158:5){C}
\tkzDrawSegments[color=red,line width=1pt](A,B A,C)
\tkzDrawPoints(A,B,C) \tkzLabelPoints(A,B,C)
\tkzProtractor[return](A,C)
\end{tikzpicture}
```

第二十章 实例分析

20.1 相似等腰三角形

以下示例选自精彩的 **Descartes et les Mathématiques** 网站，在此，未对原文进行任何修改，仅用 tkz-euclide 宏包对其进行了绘制。

https://debart.pagesperso-orange.fr/seconde/triangle.html

参考文献:

- Géométrie au Bac - Tangente, special issue no. 8 - Exercise 11, page 11
- Elisabeth Busser and Gilles Cohen: 200 nouveaux problèmes du “Monde” - POLE 2007 (200 new problems of "Le Monde")
- Affaire de logique nř 364 - Le Monde February 17, 2004

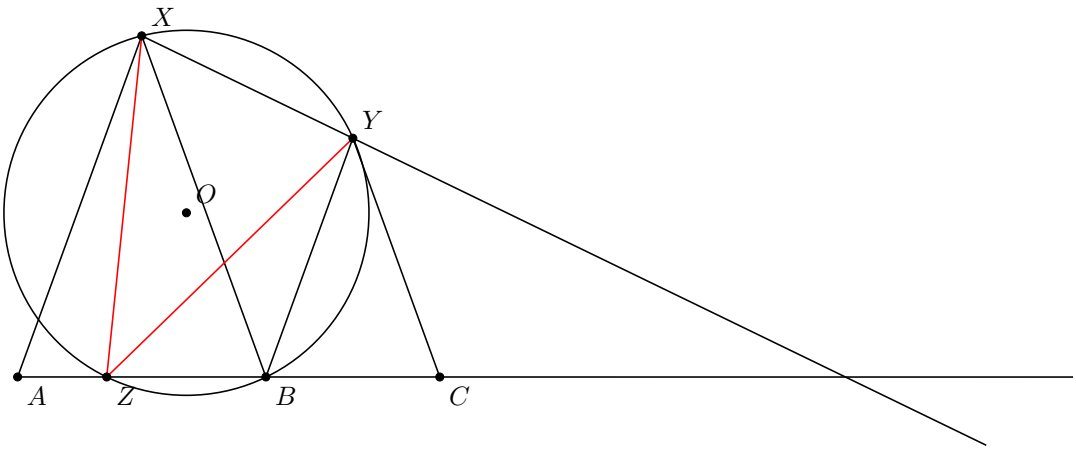
构造相似等腰三角形有两种方式，一是 *Tangente* 杂志提供的方法，另一个是 *Le Monde* 杂志提供的方法。

“Tangente” 杂志编辑: 两个相似等腰三角形 AXB 和 BYC 由主顶点分别是 X 和 Y，其中，A、B 和 C 共线。令 α 为顶点的角度 ($\widehat{AXB} = \widehat{BYC}$)，然后可以构造与前两个等腰三角形相似的第三个等腰三角形 XZY，其主顶点是 Z 点，需要证明 Z 点属于直线 (AC).

“Le Monde” 杂志编辑: 两个相似等腰三角形 AXB 和 BYC 的主顶点分别是 X 和 Y，其中 A、B 和 C 点共线。令 α 为顶点的角度 ($\widehat{AXB} = \widehat{BYC}$)，线段 $[AC]$ 上的点 Z 与 X 和 Y 的距离相等。
那么这两个顶点的角度是多少?

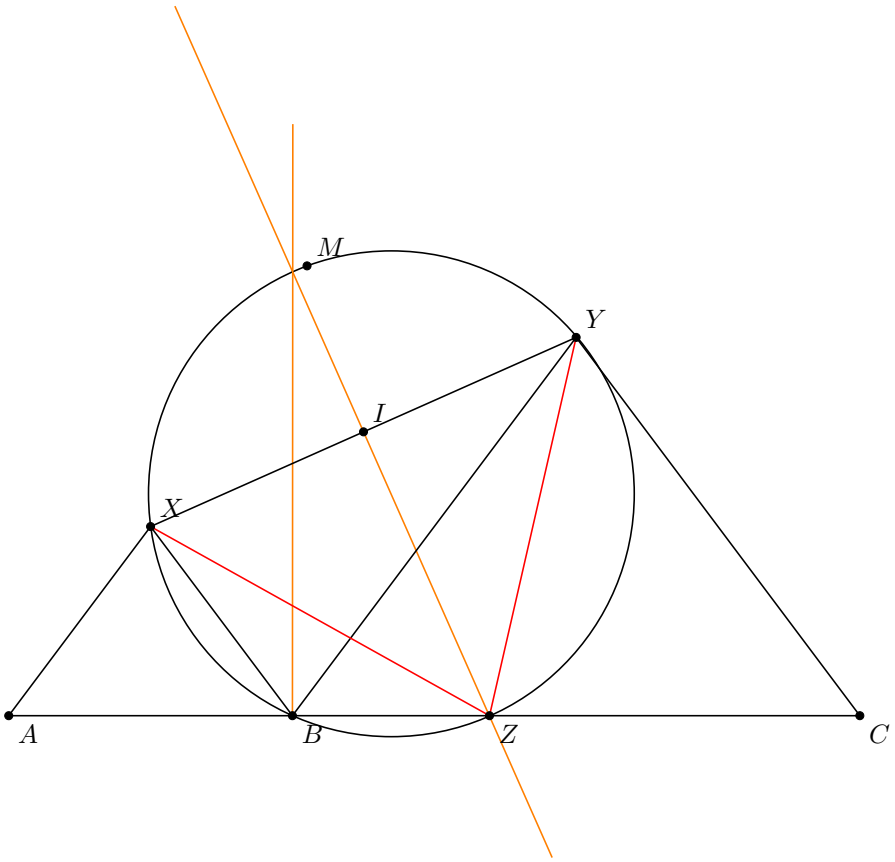
下面两页给出了解决这两个问题的 tkz-euclide 代码和绘图过程，但在查看代码前，建议先搜索相关文献。这些代码展示了这两种推导过程。

20.1.1 “Tangente” 方法




```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(6,0){X}   \tkzDefPoint(3,3){Y}
\tkzDefShiftPoint[X](-110:6){A}    \tkzDefShiftPoint[X](-70:6){B}
\tkzDefShiftPoint[Y](-110:4.2){A'} \tkzDefShiftPoint[Y](-70:4.2){B'}
\tkzDefPointBy[translation= from A' to B ](Y) \tkzGetPoint{Y}
\tkzDefPointBy[translation= from A' to B ](B') \tkzGetPoint{C}
\tkzInterLL(A,B)(X,Y) \tkzGetPoint{O}
\tkzDefMidPoint(X,Y) \tkzGetPoint{I}
\tkzDefPointWith[orthogonal](I,Y)
\tkzInterLL(I,tkzPointResult)(A,B) \tkzGetPoint{Z}
\tkzDefCircle[circum](X,Y,B) \tkzGetPoint{O}
\tkzDrawCircle(O,X)
\tkzDrawLines[add = 0 and 1.5](A,C) \tkzDrawLines[add = 0 and 3](X,Y)
\tkzDrawSegments(A,X B,X B,Y C,Y)   \tkzDrawSegments[color=red](X,Z Y,Z)
\tkzDrawPoints(A,B,C,X,Y,O,Z)
\tkzLabelPoints(A,B,C,Z)   \tkzLabelPoints[above right](X,Y,O)
\end{tikzpicture}
```

20.1.2 “Le Monde” 方法



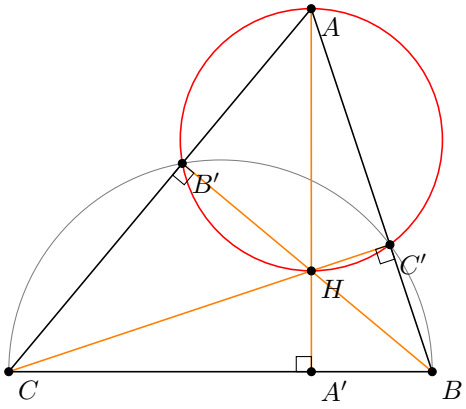

```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){A}
\tkzDefPoint(3,0){B}
\tkzDefPoint(9,0){C}
\tkzDefPoint(1.5,2){X}
\tkzDefPoint(6,4){Y}
\tkzDefCircle[circum](X,Y,B)        \tkzGetPoint{O}
\tkzDefMidPoint(X,Y)                \tkzGetPoint{I}
\tkzDefPointWith[orthogonal](I,Y)   \tkzGetPoint{i}
\tkzDrawLines[add = 2 and 1,color=orange](I,i)
\tkzInterLL(I,i)(A,B)               \tkzGetPoint{Z}
\tkzInterLC(I,i)(O,B)               \tkzGetSecondPoint{M}
\tkzDefPointWith[orthogonal](B,Z)   \tkzGetPoint{b}
\tkzDrawCircle(O,B)
\tkzDrawLines[add = 0 and 2,color=orange](B,b)
\tkzDrawSegments(A,X B,X B,Y C,Y A,C X,Y)
\tkzDrawSegments[color=red](X,Z Y,Z)
\tkzDrawPoints(A,B,C,X,Y,Z,M,I)
\tkzLabelPoints(A,B,C,Z)
\tkzLabelPoints[above right](X,Y,M,I)
\end{tikzpicture}
```

20.2 三角形的高

以下示例选自精彩的 **Descartes et les Mathématiques** 网站 (Descartes and the Mathematics)。
https://debart.pagesperso-orange.fr/geoplan/geometrie_triangle.html

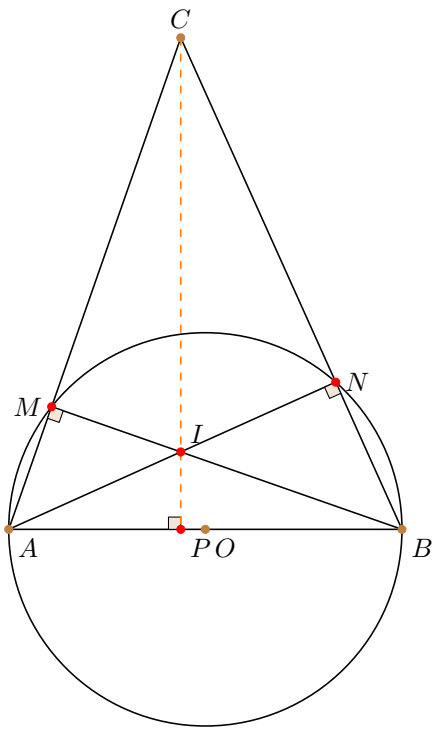
20.2.1 构造方式一

三条高相交于 H 点。




```
\begin{tikzpicture}[scale=.8]
\tkzDefPoint(0,0){C} \tkzDefPoint(7,0){B} \tkzDefPoint(5,6){A}
\tkzDrawPolygon(A,B,C)
\tkzDefMidPoint(C,B) \tkzGetPoint{I}
\tkzDrawArc(I,B)(C)
\tkzInterLC(A,C)(I,B) \tkzGetSecondPoint{B'}
\tkzInterLC(A,B)(I,B) \tkzGetFirstPoint{C'}
\tkzInterLL(B,B')(C,C') \tkzGetPoint{H}
\tkzInterLL(A,H)(C,B) \tkzGetPoint{A'}
\tkzDefCircle[circum](A,B',C') \tkzGetPoint{O}
\tkzDrawCircle[color=red](O,A)
\tkzDrawSegments[color=orange](B,B' C,C' A,A')
\tkzMarkRightAngles(C,B',B B,C',C C,A',A)
\tkzDrawPoints(A,B,C,A',B',C',H)
\tkzLabelPoints(A,B,C,A',B',C',H)
\end{tikzpicture}
```

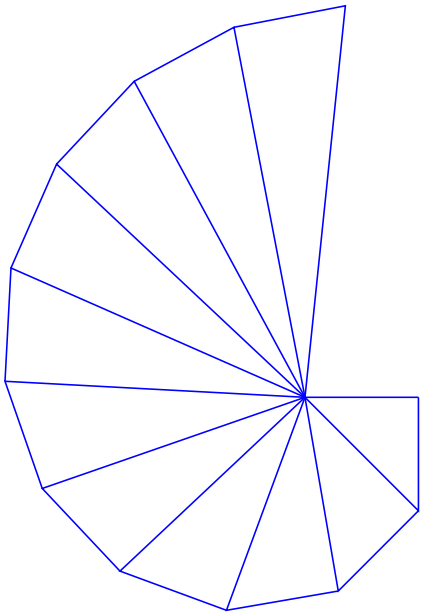
20.2.2 构造方式二




```
\begin{tikzpicture}[scale=0.65]
\tkzDefPoint(0,0){A} \tkzDefPoint(8,0){B} \tkzDefPoint(3.5,10){C}
\tkzDefMidPoint(A,B) \tkzGetPoint{O}
\tkzDefPointBy[projection=onto A-B](C) \tkzGetPoint{P}
\tkzInterLC(C,A)(O,A) \tkzGetSecondPoint{M}
\tkzInterLC(C,B)(O,A) \tkzGetFirstPoint{N}
\tkzInterLL(B,M)(A,N) \tkzGetPoint{I}
\tkzDrawCircle[diameter](A,B)
\tkzDrawSegments(C,A C,B A,B B,M A,N)
\tkzMarkRightAngles[fill=brown!20](A,M,B A,N,B A,P,C)
\tkzDrawSegment[style=dashed,color=orange](C,P)
\tkzLabelPoints(O,A,B,P)
\tkzLabelPoint[left](M){$M$}
\tkzLabelPoint[right](N){$N$}
\tkzLabelPoint[above](C){$C$}
\tkzLabelPoint[above right](I){$I$}
\tkzDrawPoints[color=red](M,N,P,I)
\tkzDrawPoints[color=brown](O,A,B,C)
\end{tikzpicture}
```

20.3 整数的算术平方根

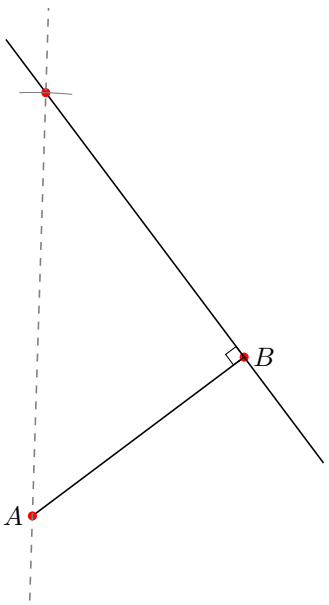
本例演示了如何用尺规求解 1、$\sqrt{2}$ 和 $\sqrt{3}$ 的方法。



```
\begin{tikzpicture}[scale=1.5]
\tkzDefPoint(0,0){O}
\tkzDefPoint(1,0){a0}
\tkzDrawSegment[blue](O,a0)
\foreach \i [count=\j] in {0,...,10}{%
\tkzDefPointWith[orthogonal normed](a\i,O)
\tkzGetPoint{a\j}
\tkzDrawPolySeg[color=blue](a\i,a\j,O)}
\end{tikzpicture}
```

20.4 直角三角

有一线段 $[AB]$，然后确定一点 C 使 $AC = 8\,\text{cm}$，并且 ABC 是以 B 为直角的直角三角形。

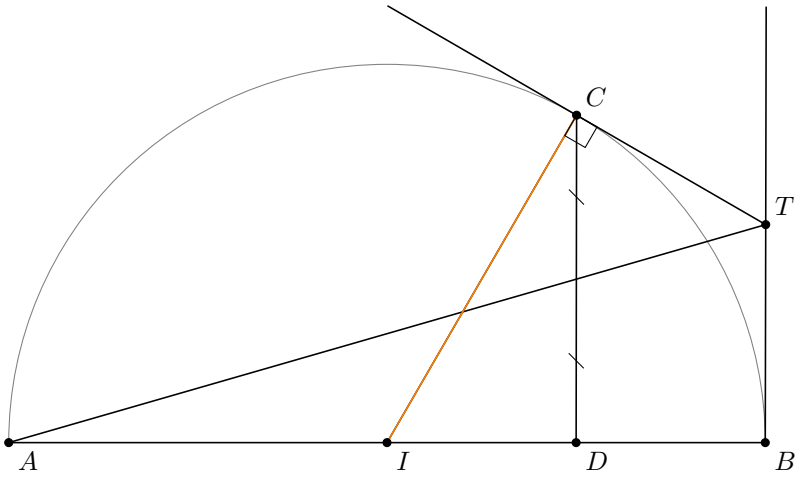



```
\begin{tikzpicture}[scale=.7]
\tkzDefPoint["$A$" left](2,1){A}
\tkzDefPoint["$ B $" right](6,4){B}
\tkzDrawSegment(A,B)
\tkzDrawPoint[color=red](A)
\tkzDrawPoint[color=red](B)
\tkzDefPointWith[orthogonal,K=-1](B,A)
\tkzDrawLine[add = .5 and .5](B,tkzPointResult)
\tkzInterLC[R](B,tkzPointResult)(A,8 cm)
\tkzGetPoints{C}{J}
\tkzDrawPoint[color=red](C)
\tkzCompass(A,C)
\tkzMarkRightAngle(A,B,C)
\tkzDrawLine[color=gray,style=dashed](A,C)
\end{tikzpicture}
```

20.5 阿基米德等分

这是伟大的希腊数学家阿基米德证明的一个古老问题。下图有一个直径为 AB 的半圆，一条直线在 B 点与半圆相切，在 C 点有半圆的另一条切线。把 C 点投影到线段 $[AB]$ 上的 D 点。两条切线相交于 T 点。

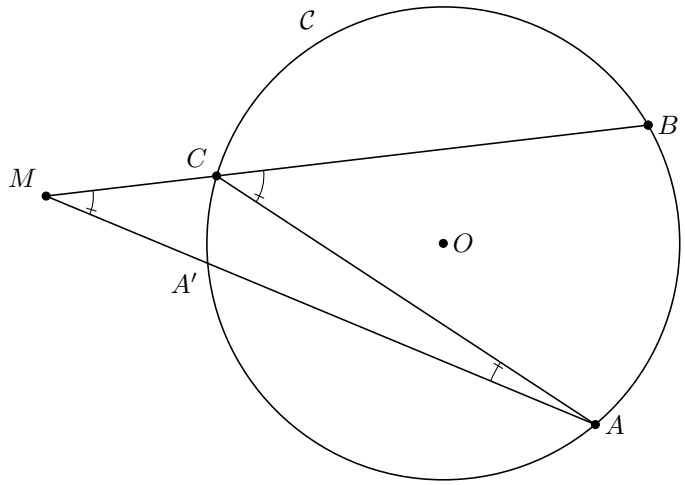
现证明直线 (AT) 平分直线 (CD)




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){A}\tkzDefPoint(6,0){D}
\tkzDefPoint(8,0){B}\tkzDefPoint(4,0){I}
\tkzDefLine[orthogonal=through D](A,D)
\tkzInterLC[R](D,tkzPointResult)(I,4 cm) \tkzGetFirstPoint{C}
\tkzDefLine[orthogonal=through C](I,C)   \tkzGetPoint{c}
\tkzDefLine[orthogonal=through B](A,B)   \tkzGetPoint{b}
\tkzInterLL(C,c)(B,b) \tkzGetPoint{T}
\tkzInterLL(A,T)(C,D) \tkzGetPoint{P}
\tkzDrawArc(I,B)(A)
\tkzDrawSegments(A,B A,T C,D I,C) \tkzDrawSegment[color=orange](I,C)
\tkzDrawLine[add = 1 and 0](C,T)  \tkzDrawLine[add = 0 and 1](B,T)
\tkzMarkRightAngle(I,C,T)
\tkzDrawPoints(A,B,I,D,C,T)
\tkzLabelPoints(A,B,I,D)  \tkzLabelPoints[above right](C,T)
\tkzMarkSegment[pos=.25,mark=s|](C,D) \tkzMarkSegment[pos=.75,mark=s|](C,D)
\end{tikzpicture}
```

20.6 示例 (Dimitris Kapeta)

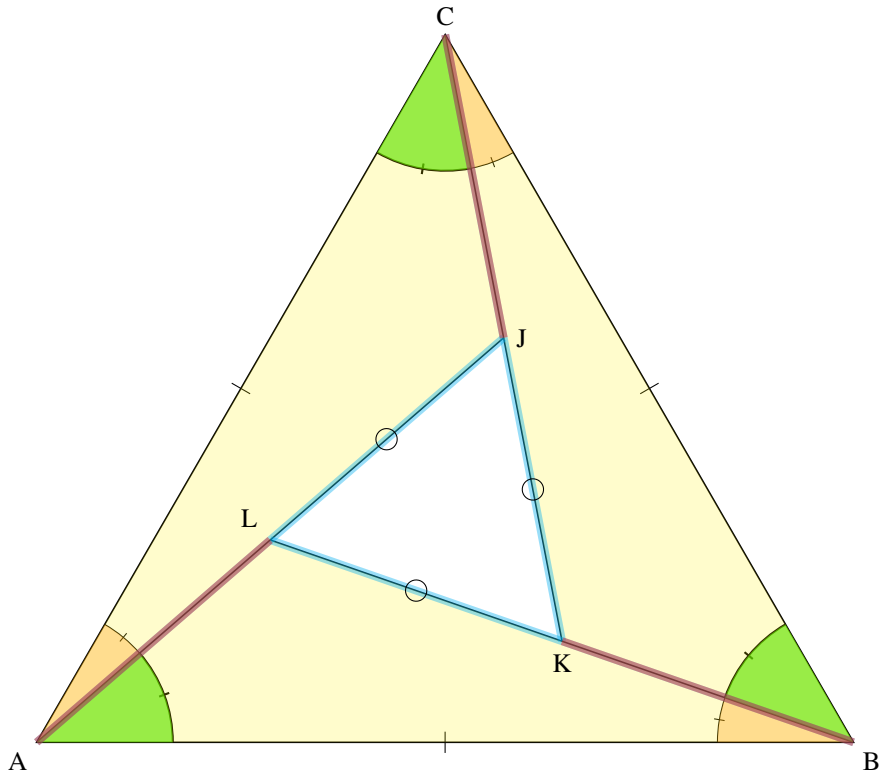
在本例中，由于 $\widehat{CAM}$ 过小，因此需要在\tkzMarkAngle 命令中使用 mkpos=.2 选项。另一种解决方案是使用\tkzFillAngle 命令。




```
\begin{tikzpicture}[scale=1.25]
\tkzDefPoint(0,0){O}
\tkzDefPoint(2.5,0){N}
\tkzDefPoint(-4.2,0.5){M}
\tkzDefPointBy[rotation=center O angle 30](N)
\tkzGetPoint{B}
\tkzDefPointBy[rotation=center O angle -50](N)
\tkzGetPoint{A}
\tkzInterLC(M,B)(O,N) \tkzGetFirstPoint{C}
\tkzInterLC(M,A)(O,N) \tkzGetSecondPoint{A'}
\tkzMarkAngle[mkpos=.2, size=0.5](A,C,B)
\tkzMarkAngle[mkpos=.2, size=0.5](A,M,C)
\tkzDrawSegments(A,C M,A M,B)
\tkzDrawCircle(O,N)
\tkzLabelCircle[above left](O,N)(120){$\mathcal{C}$}
\tkzMarkAngle[mkpos=.2, size=1.2](C,A,M)
\tkzDrawPoints(O, A, B, M, B, C)
\tkzLabelPoints[right](O,A,B)
\tkzLabelPoints[above left](M,C)
\tkzLabelPoint[below left](A'){$A'$}
\end{tikzpicture}
```

20.7 证明示例 1(John Kitzmiller)

证明 $\triangle LKJ$ 是等边三角形。

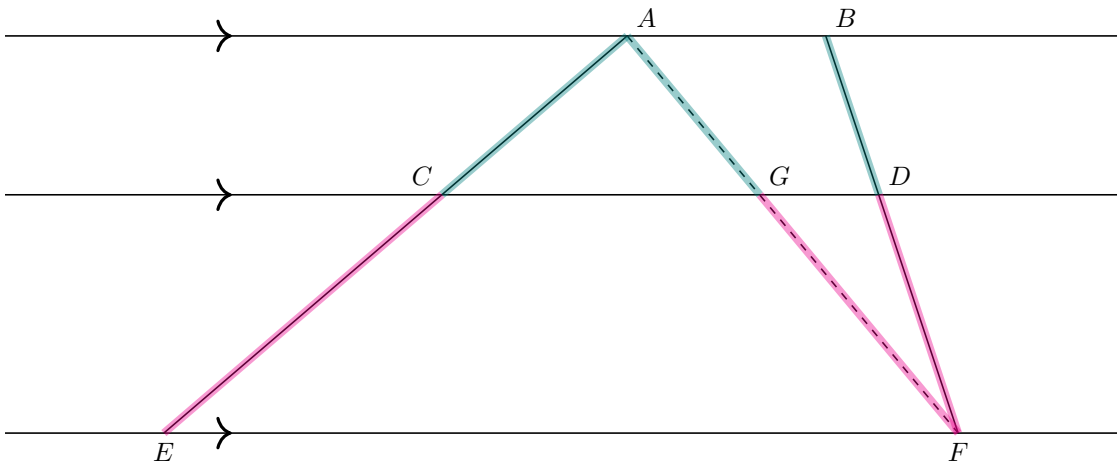



```
\begin{tikzpicture}[scale=1.8]
\tkzDefPoint[label=below left:A](0,0){A} \tkzDefPoint[label=below right:B](6,0){B}
\tkzDefTriangle[equilateral](A,B) \tkzGetPoint{C}
\tkzMarkSegments[mark=|](A,B A,C B,C)
\tkzDefBarycentricPoint(A=1,B=2) \tkzGetPoint{C'}
\tkzDefBarycentricPoint(A=2,C=1) \tkzGetPoint{B'}
\tkzDefBarycentricPoint(C=2,B=1) \tkzGetPoint{A'}
\tkzInterLL(A,A')(C,C') \tkzGetPoint{J}
\tkzInterLL(C,C')(B,B') \tkzGetPoint{K}
\tkzInterLL(B,B')(A,A') \tkzGetPoint{L}
\tkzLabelPoint[above](C){C}
\tkzDrawPolygon(A,B,C) \tkzDrawSegments(A,J B,L C,K)
\tkzMarkAngles[size=1 cm](J,A,C K,C,B L,B,A)
\tkzMarkAngles[thick,size=1 cm](A,C,J C,B,K B,A,L)
\tkzMarkAngles[opacity=.5](A,C,J C,B,K B,A,L)
\tkzFillAngles[fill= orange,size=1 cm,opacity=.3](J,A,C K,C,B L,B,A)
\tkzFillAngles[fill=orange, opacity=.3,thick,size=1,](A,C,J C,B,K B,A,L)
\tkzFillAngles[fill=green, size=1, opacity=.5](A,C,J C,B,K B,A,L)
\tkzFillPolygon[color=yellow, opacity=.2](J,A,C)
\tkzFillPolygon[color=yellow, opacity=.2](K,B,C)
\tkzFillPolygon[color=yellow, opacity=.2](L,A,B)
\tkzDrawSegments[line width=3pt,color=cyan,opacity=0.4](A,J C,K B,L)
\tkzDrawSegments[line width=3pt,color=red,opacity=0.4](A,L B,K C,J)
\tkzMarkSegments[mark=o](J,K K,L L,J)
\tkzLabelPoint[right](J){J} \tkzLabelPoint[below](K){K}
\tkzLabelPoint[above left](L){L}
\end{tikzpicture}
```

20.8 证明示例 2(John Kitzmiller)

证明：$\dfrac{AC}{CE} = \dfrac{BD}{DF}$

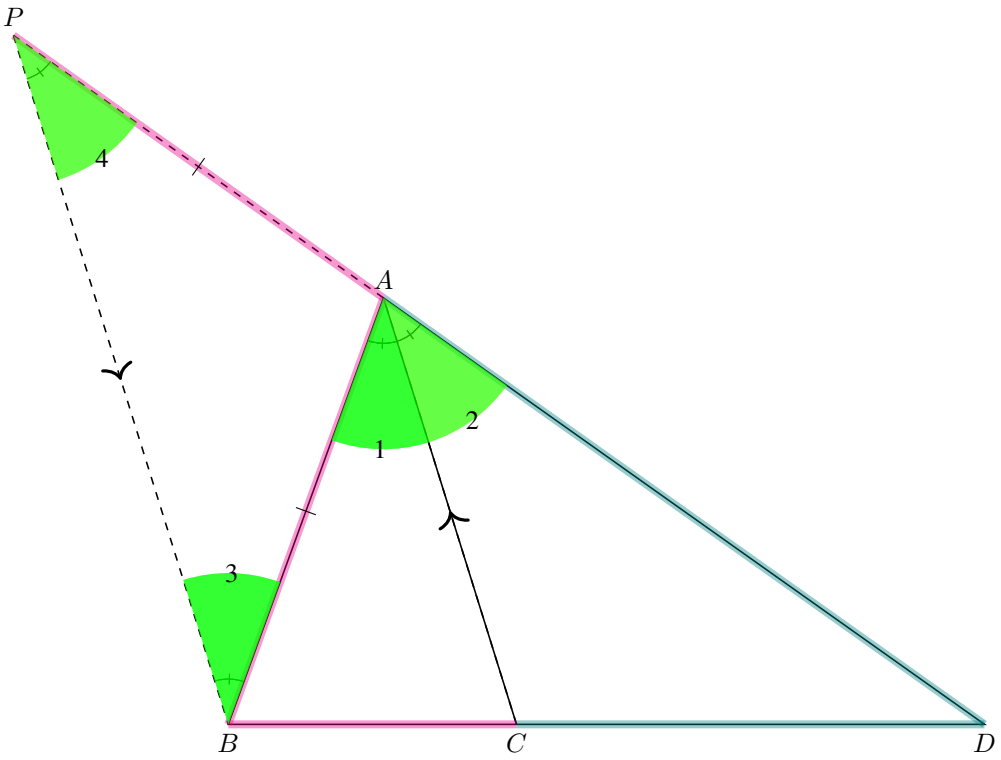
John 的另一个有趣的例子是如何在 tkz-euclide 宏包中使用 TikZ 的类似 decoration 和 postaction 选项。




```
\begin{tikzpicture}[scale=1.75,decoration={markings,
mark=at position 3cm with {\arrow[scale=2]{>}}}]
\tkzDefPoints{0/0/E, 6/0/F, 0/1.8/P, 6/1.8/Q, 0/3/R, 6/3/S}
\tkzDrawLines[postaction={decorate}](E,F P,Q R,S)
\tkzDefPoints{3.5/3/A, 5/3/B}
\tkzDrawSegments(E,A F,B)
\tkzInterLL(E,A)(P,Q) \tkzGetPoint{C}
\tkzInterLL(B,F)(P,Q) \tkzGetPoint{D}
\tkzLabelPoints[above right](A,B)
\tkzLabelPoints[below](E,F)
\tkzLabelPoints[above left](C)
\tkzDrawSegments[style=dashed](A,F)
\tkzInterLL(A,F)(P,Q) \tkzGetPoint{G}
\tkzLabelPoints[above right](D,G)
\tkzDrawSegments[color=teal, line width=3pt, opacity=0.4](A,C A,G)
\tkzDrawSegments[color=magenta, line width=3pt, opacity=0.4](C,E G,F)
\tkzDrawSegments[color=teal, line width=3pt, opacity=0.4](B,D)
\tkzDrawSegments[color=magenta, line width=3pt, opacity=0.4](D,F)
\end{tikzpicture}
```

20.9 证明示例 3(John Kitzmiller)

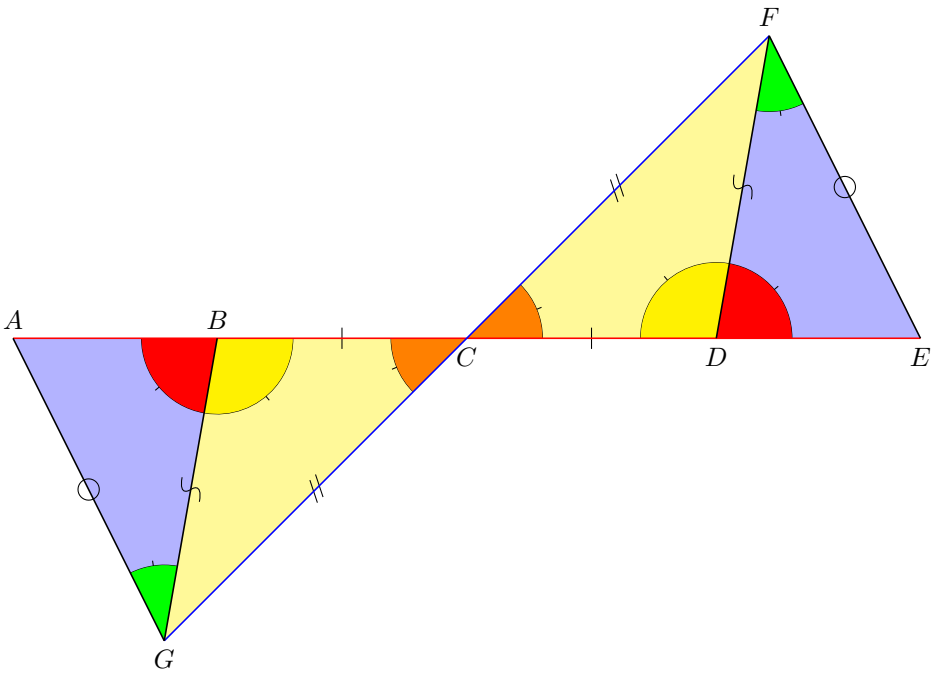
证明：$\dfrac{BC}{CD} = \dfrac{AB}{AD}$ (角平分线).




```
\begin{tikzpicture}[scale=2]
\tkzDefPoints{0/0/B, 5/0/D}   \tkzDefPoint(70:3){A}
\tkzDrawPolygon(B,D,A)
\tkzDefLine[bisector](B,A,D)  \tkzGetPoint{a}
\tkzInterLL(A,a)(B,D)         \tkzGetPoint{C}
\tkzDefLine[parallel=through B](A,C) \tkzGetPoint{b}
\tkzInterLL(A,D)(B,b)         \tkzGetPoint{P}
\begin{scope}[decoration={markings, mark=at position .5 with {\arrow[scale=2]{>}}}]
\tkzDrawSegments[postaction={decorate},dashed](C,A P,B)
\end{scope}
\tkzDrawSegment(A,C) \tkzDrawSegment[style=dashed](A,P)
\tkzLabelPoints[below](B,C,D) \tkzLabelPoints[above](A,P)
\tkzDrawSegments[color=magenta, line width=3pt, opacity=0.4](B,C P,A)
\tkzDrawSegments[color=teal,    line width=3pt, opacity=0.4](C,D A,D)
\tkzDrawSegments[color=magenta, line width=3pt, opacity=0.4](A,B)
\tkzMarkAngles[size=3mm](B,A,C C,A,D)
\tkzMarkAngles[size=3mm](B,A,C A,B,P)
\tkzMarkAngles[size=3mm](B,P,A C,A,D)
\tkzMarkAngles[size=3mm](B,A,C A,B,P B,P,A C,A,D)
\tkzFillAngles[fill=green,  opacity=0.5](B,A,C A,B,P)
\tkzFillAngles[fill=yellow, opacity=0.3](B,P,A C,A,D)
\tkzFillAngles[fill=green,  opacity=0.6](B,A,C A,B,P B,P,A C,A,D)
\tkzLabelAngle[pos=1](B,A,C){1}  \tkzLabelAngle[pos=1](C,A,D){2}
\tkzLabelAngle[pos=1](A,B,P){3}  \tkzLabelAngle[pos=1](B,P,A){4}
\tkzMarkSegments[mark=|](A,B A,P)
\end{tikzpicture}
```

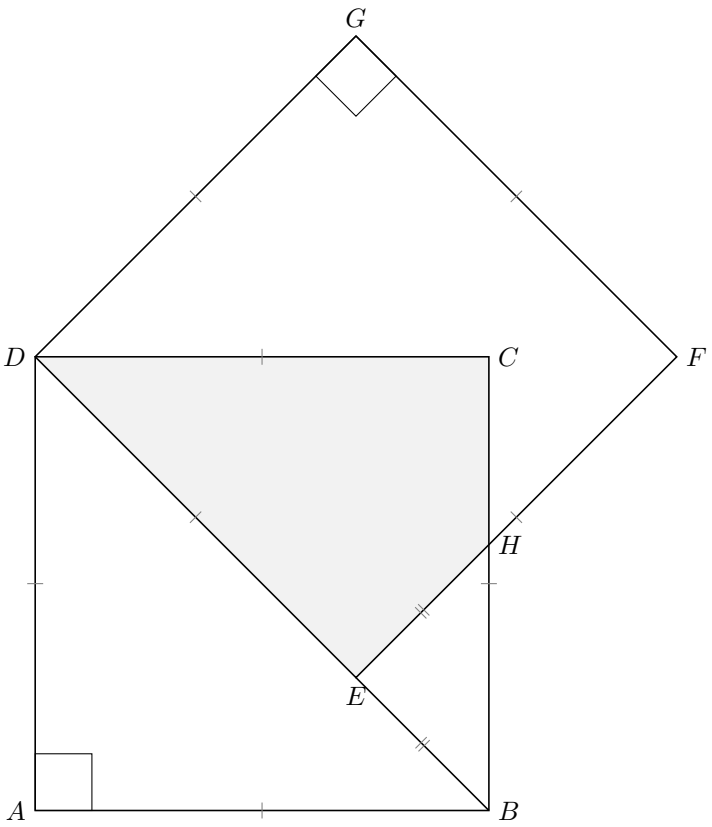
20.10 证明示例 4(John Kitzmiller)

证明：$\overline{AG} \cong \overline{EF}$ (Detour).



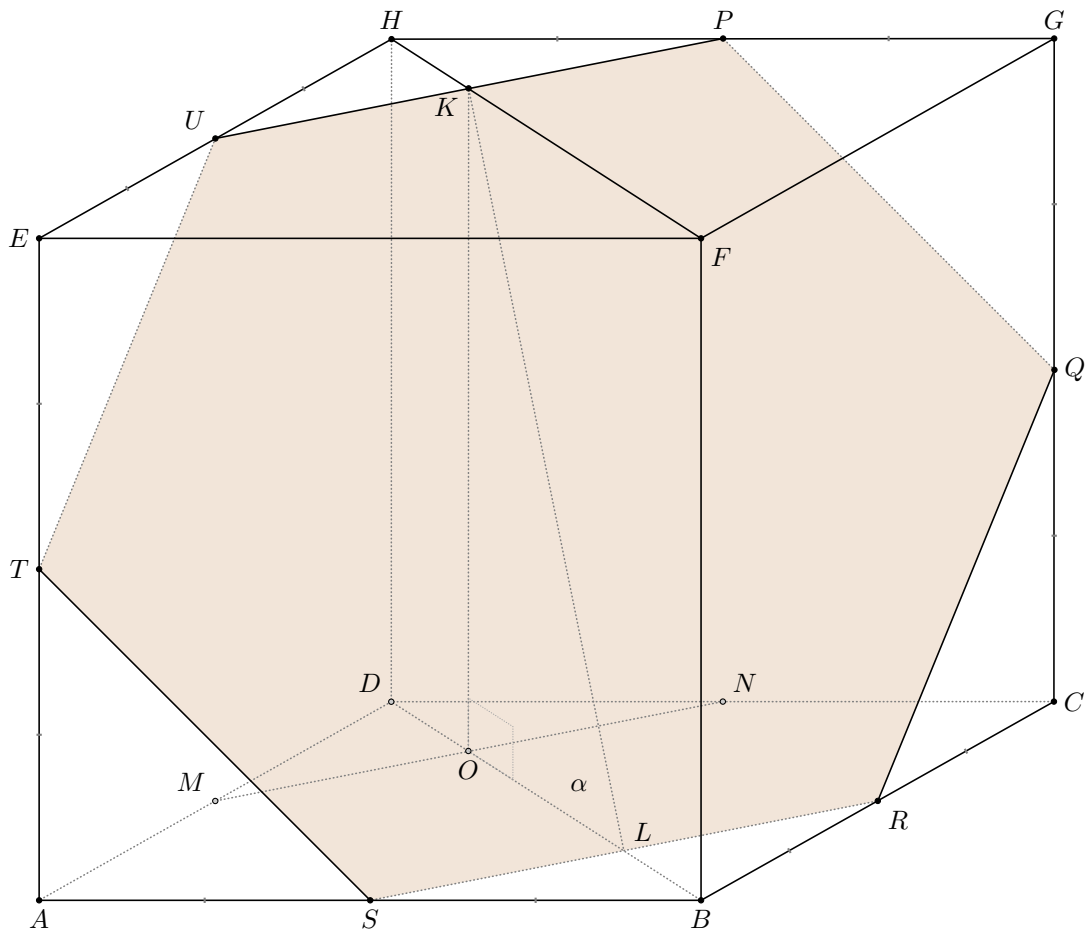

```
\begin{tikzpicture}[scale=2]
\tkzDefPoint(0,3){A}    \tkzDefPoint(6,3){E}  \tkzDefPoint(1.35,3){B}
\tkzDefPoint(4.65,3){D} \tkzDefPoint(1,1){G}  \tkzDefPoint(5,5){F}
\tkzDefMidPoint(A,E)    \tkzGetPoint{C}
\tkzFillPolygon[yellow, opacity=0.4](B,G,C)
\tkzFillPolygon[yellow, opacity=0.4](D,F,C)
\tkzFillPolygon[blue, opacity=0.3](A,B,G)
\tkzFillPolygon[blue, opacity=0.3](E,D,F)
\tkzMarkAngles[size=0.5 cm](B,G,A D,F,E)
\tkzMarkAngles[size=0.5 cm](B,C,G D,C,F)
\tkzMarkAngles[size=0.5 cm](G,B,C F,D,C)
\tkzMarkAngles[size=0.5 cm](A,B,G E,D,F)
\tkzFillAngles[size=0.5 cm,fill=green](B,G,A D,F,E)
\tkzFillAngles[size=0.5 cm,fill=orange](B,C,G D,C,F)
\tkzFillAngles[size=0.5 cm,fill=yellow](G,B,C F,D,C)
\tkzFillAngles[size=0.5 cm,fill=red](A,B,G E,D,F)
\tkzMarkSegments[mark=|](B,C D,C)  \tkzMarkSegments[mark=s||](G,C F,C)
\tkzMarkSegments[mark=o](A,G E,F)  \tkzMarkSegments[mark=s](B,G D,F)
\tkzDrawSegment[color=red](A,E)
\tkzDrawSegment[color=blue](F,G)
\tkzDrawSegments(A,G G,B E,F F,D)
\tkzLabelPoints[below](C,D,E,G)  \tkzLabelPoints[above](A,B,F)
\end{tikzpicture}
```

20.11 构图示例 1(Indonesia)



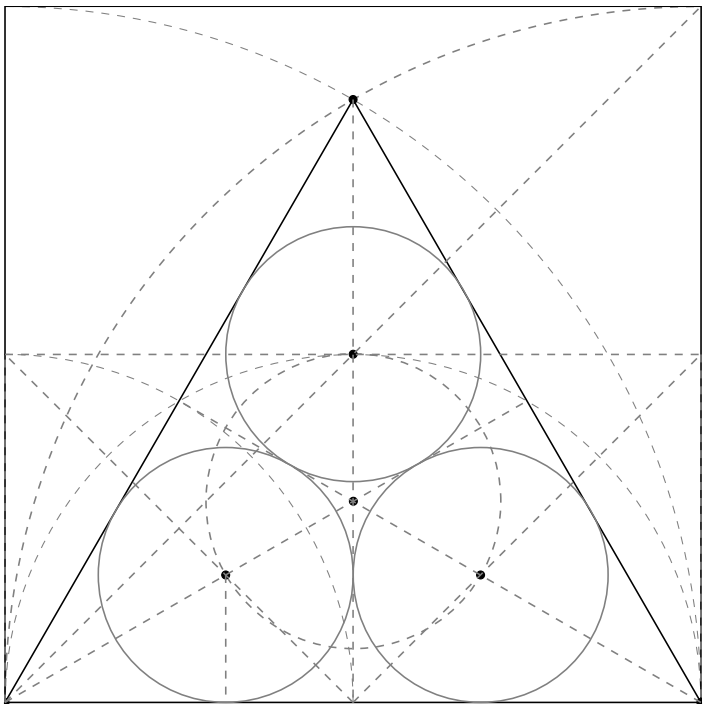

```
\begin{tikzpicture}[scale=3]
\tkzDefPoints{0/0/A,2/0/B}
\tkzDefSquare(A,B) \tkzGetPoints{C}{D}
\tkzDefPointBy[rotation=center D angle 45](C)\tkzGetPoint{G}
\tkzDefSquare(G,D)\tkzGetPoints{E}{F}
\tkzInterLL(B,C)(E,F)\tkzGetPoint{H}
\tkzFillPolygon[gray!10](D,E,H,C,D)
\tkzDrawPolygon(A,...,D)\tkzDrawPolygon(D,...,G)
\tkzDrawSegment(B,E)
\tkzMarkSegments[mark=|,size=3pt,color=gray](A,B B,C C,D D,A E,F F,G G,D D,E)
\tkzMarkSegments[mark=||,size=3pt,color=gray](B,E E,H)
\tkzLabelPoints[left](A,D)
\tkzLabelPoints[right](B,C,F,H)
\tkzLabelPoints[above](G)\tkzLabelPoints[below](E)
\tkzMarkRightAngles(D,A,B D,G,F)
\end{tikzpicture}
```

20.12 构图示例 2(Indonesia)



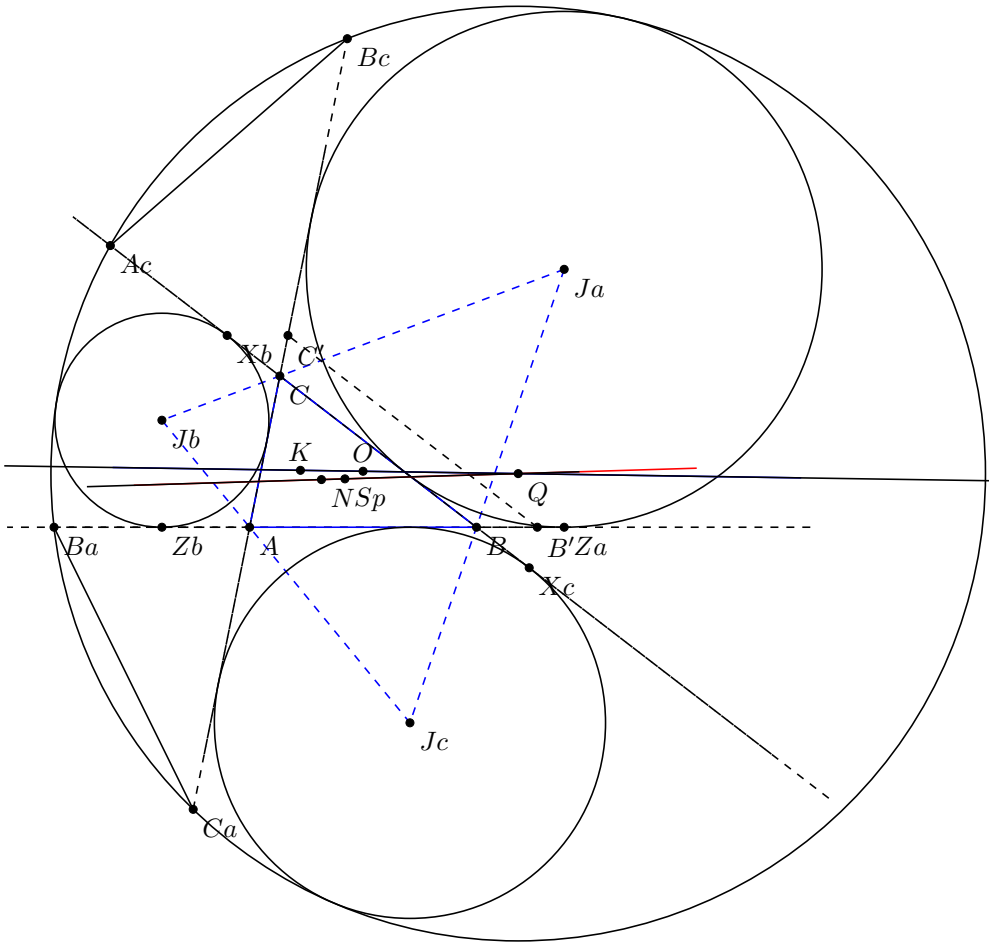
```
\begin{tikzpicture}[pol/.style={fill=brown!40,opacity=.5},
seg/.style={tkzdotted,color=gray},
hidden pt/.style={fill=gray!40},
mra/.style={color=gray!70,tkzdotted,/tkzrightangle/size=.2},
scale=3.5]
\tkzSetUpPoint[size=2]
\tkzDefPoints{0/0/A,2.5/0/B,1.33/0.75/D,0/2.5/E,2.5/2.5/F}
\tkzDefLine[parallel=through D](A,B) \tkzGetPoint{I1}
\tkzDefLine[parallel=through B](A,D) \tkzGetPoint{I2}
\tkzInterLL(D,I1)(B,I2) \tkzGetPoint{C}
\tkzDefLine[parallel=through E](A,D) \tkzGetPoint{I3}
\tkzDefLine[parallel=through D](A,E) \tkzGetPoint{I4}
\tkzInterLL(E,I3)(D,I4) \tkzGetPoint{H}
\tkzDefLine[parallel=through F](E,H) \tkzGetPoint{I5}
\tkzDefLine[parallel=through H](E,F) \tkzGetPoint{I6}
\tkzInterLL(F,I5)(H,I6) \tkzGetPoint{G}
\tkzDefMidPoint(G,H)   \tkzGetPoint{P}
\tkzDefMidPoint(G,C)   \tkzGetPoint{Q}
\tkzDefMidPoint(B,C)   \tkzGetPoint{R}
\tkzDefMidPoint(A,B)   \tkzGetPoint{S}
\tkzDefMidPoint(A,E)   \tkzGetPoint{T}
\tkzDefMidPoint(E,H)   \tkzGetPoint{U}
\tkzDefMidPoint(A,D)   \tkzGetPoint{M}
\tkzDefMidPoint(D,C)   \tkzGetPoint{N}
\tkzInterLL(B,D)(S,R)  \tkzGetPoint{L}
\tkzInterLL(H,F)(U,P)  \tkzGetPoint{K}
\tkzDefLine[parallel=through K](D,H) \tkzGetPoint{I7}
\tkzInterLL(K,I7)(B,D) \tkzGetPoint{O}
\tkzFillPolygon[pol](P,Q,R,S,T,U)
\tkzDrawSegments[seg](K,O K,L P,Q R,S T,U C,D H,D A,D M,N B,D)
\tkzDrawSegments(E,H B,C G,F G,H G,C Q,R S,T U,P H,F)
\tkzDrawPolygon(A,B,F,E)
\tkzDrawPoints(A,B,C,E,F,G,H,P,Q,R,S,T,U,K)
\tkzDrawPoints[hidden pt](M,N,O,D)
\tkzMarkRightAngle[mra](L,O,K)
\tkzMarkSegments[mark=|,size=1pt,thick,color=gray](A,S B,S B,R C,R Q,C Q,G G,P H,P E,U H,U E,T A,T)
\tkzLabelAngle[pos=.3](K,L,O){$\alpha$}
\tkzLabelPoints[below](O,A,S,B)
\tkzLabelPoints[above](H,P,G)
\tkzLabelPoints[left](T,E)
\tkzLabelPoints[right](C,Q)
\tkzLabelPoints[above left](U,D,M)
\tkzLabelPoints[above right](L,N)
\tkzLabelPoints[below right](F,R)
\tkzLabelPoints[below left](K)
\end{tikzpicture}
```

20.13 三个相切圆



```
\begin{tikzpicture}[scale=1.15]
\tkzDefPoints{0/0/A,8/0/B,0/4/a,8/4/b,8/8/c}
\tkzDefTriangle[equilateral](A,B) \tkzGetPoint{C}
\tkzDrawPolygon(A,B,C)
\tkzDefSquare(A,B) \tkzGetPoints{D}{E}
\tkzClipBB
\tkzDefMidPoint(A,B) \tkzGetPoint{M}
\tkzDefMidPoint(B,C) \tkzGetPoint{N}
\tkzDefMidPoint(A,C) \tkzGetPoint{P}
\tkzDrawSemiCircle[gray,dashed](M,B)
\tkzDrawSemiCircle[gray,dashed](A,M)
\tkzDrawSemiCircle[gray,dashed](A,B)
\tkzDrawCircle[gray,dashed](B,A)
\tkzInterLL(A,N)(M,a) \tkzGetPoint{Ia}
\tkzDefPointBy[projection = onto A-B](Ia) \tkzGetPoint{ha}
\tkzDrawCircle[gray](Ia,ha)
\tkzInterLL(B,P)(M,b) \tkzGetPoint{Ib}
\tkzDefPointBy[projection = onto A-B](Ib) \tkzGetPoint{hb}
\tkzDrawCircle[gray](Ib,hb)
\tkzInterLL(A,c)(M,C) \tkzGetPoint{Ic}
\tkzDefPointBy[projection = onto A-C](Ic) \tkzGetPoint{hc}
\tkzDrawCircle[gray](Ic,hc)
\tkzInterLL(A,Ia)(B,Ib) \tkzGetPoint{G}
\tkzDrawCircle[gray,dashed](G,Ia)
\tkzDrawPolySeg(A,E,D,B)
\tkzDrawPoints(A,B,C) \tkzDrawPoints(G,Ia,Ib,Ic)
\tkzDrawSegments[gray,dashed](C,M A,N B,P M,a M,b A,a a,b b,B A,D Ia,ha)
\end{tikzpicture}
```

20.14 APOLLONIUS 圆



```
\begin{tikzpicture}[scale=.5]
\tkzDefPoints{0/0/A,6/0/B,0.8/4/C}
\tkzDefTriangleCenter[euler](A,B,C)        \tkzGetPoint{N}
\tkzDefTriangleCenter[circum](A,B,C)       \tkzGetPoint{O}
\tkzDefTriangleCenter[lemoine](A,B,C)      \tkzGetPoint{K}
\tkzDefTriangleCenter[spieker](A,B,C)      \tkzGetPoint{Sp}
\tkzDefExCircle(A,B,C)     \tkzGetPoint{Jb}
\tkzDefExCircle(C,A,B)     \tkzGetPoint{Ja}
\tkzDefExCircle(B,C,A)     \tkzGetPoint{Jc}
\tkzDefPointBy[projection=onto B-C ](Jc)   \tkzGetPoint{Xc}
\tkzDefPointBy[projection=onto B-C ](Jb)   \tkzGetPoint{Xb}
\tkzDefPointBy[projection=onto A-B ](Ja)   \tkzGetPoint{Za}
\tkzDefPointBy[projection=onto A-B ](Jb)   \tkzGetPoint{Zb}
\tkzDefLine[parallel=through Xc](A,C)       \tkzGetPoint{X'c}
\tkzDefLine[parallel=through Xb](A,B)       \tkzGetPoint{X'b}
\tkzDefLine[parallel=through Za](C,A)       \tkzGetPoint{Z'a}
\tkzDefLine[parallel=through Zb](C,B)       \tkzGetPoint{Z'b}
\tkzInterLL(Xc,X'c)(A,B)                    \tkzGetPoint{B'}
\tkzInterLL(Xb,X'b)(A,C)                    \tkzGetPoint{C'}
\tkzInterLL(Za,Z'a)(C,B)                    \tkzGetPoint{A''}
\tkzInterLL(Zb,Z'b)(C,A)                    \tkzGetPoint{B''}
\tkzDefPointBy[reflection= over Jc-Jb](B') \tkzGetPoint{Ca}
\tkzDefPointBy[reflection= over Jc-Jb](C') \tkzGetPoint{Ba}
\tkzDefPointBy[reflection= over Ja-Jb](A'')\tkzGetPoint{Bc}
\tkzDefPointBy[reflection= over Ja-Jb](B'')\tkzGetPoint{Ac}
\tkzDefCircle[circum](Ac,Ca,Ba)             \tkzGetPoint{Q}
\tkzDrawCircle[circum](Ac,Ca,Ba)
\tkzDefPointWith[linear,K=1.1](Q,Ac)        \tkzGetPoint{nAc}
\tkzClipCircle[through](Q,nAc)
\tkzDrawLines[add=1.5 and 1.5,dashed](A,B B,C A,C)
\tkzDrawPolygon[color=blue](A,B,C)
\tkzDrawPolygon[dashed,color=blue](Ja,Jb,Jc)
\tkzDrawCircles[ex](A,B,C B,C,A C,A,B)
\tkzDrawLines[add=0 and 0,dashed](Ca,Bc B,Za A,Ba B',C')
\tkzDrawLine[add=1 and 1,dashed](Xb,Xc)
\tkzDrawLine[add=7 and 3,blue](O,K)
\tkzDrawLine[add=8 and 15,red](N,Sp)
\tkzDrawLines[add=10 and 10](K,O N,Sp)
\tkzDrawSegments(Ba,Ca Bc,Ac)
\tkzDrawPoints(A,B,C,N,Ja,Jb,Jc,Xb,Xc,B',C',Za,Zb,Ba,Ca,Bc,Ac,Q,Sp,K,O)
\tkzLabelPoints(A,B,C,N,Ja,Jb,Jc,Xb,Xc,B',C',Za,Zb,Ba,Ca,Bc,Ac,Q,Sp)
\tkzLabelPoints[above](K,O)
\end{tikzpicture}
```

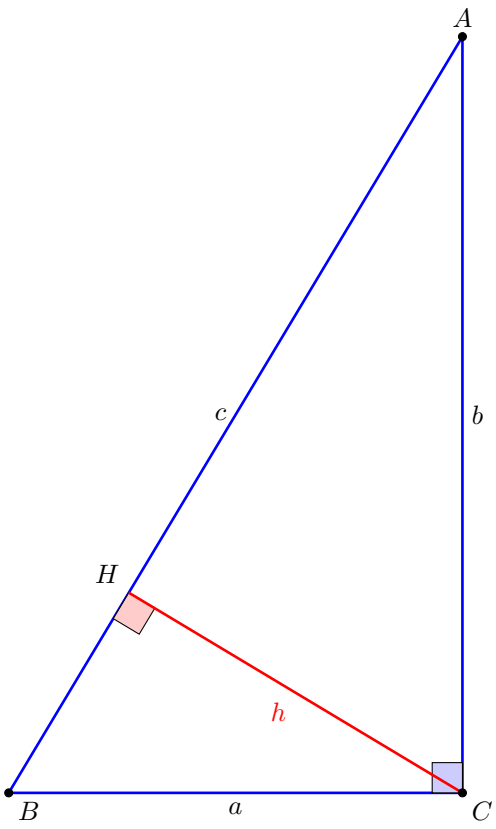
第二十一章 个性化设置

21.1 设置线条样式

\tkzSetUpLine[选项]

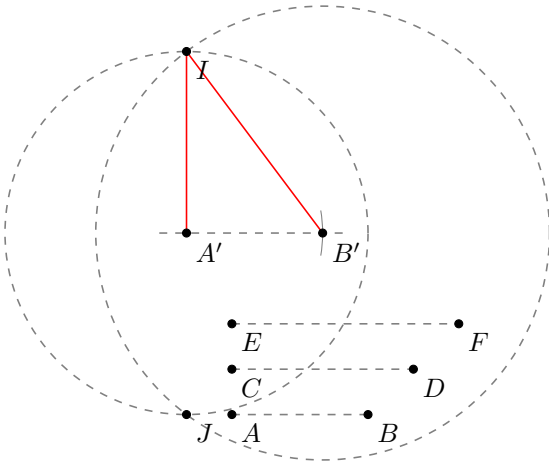
选项	默认值	含义
color	black	颜色
line width	0.4pt	线宽
style	solid	线型
add	.2 and .2	线段两端延伸长度

21.1.1 改变线宽示例



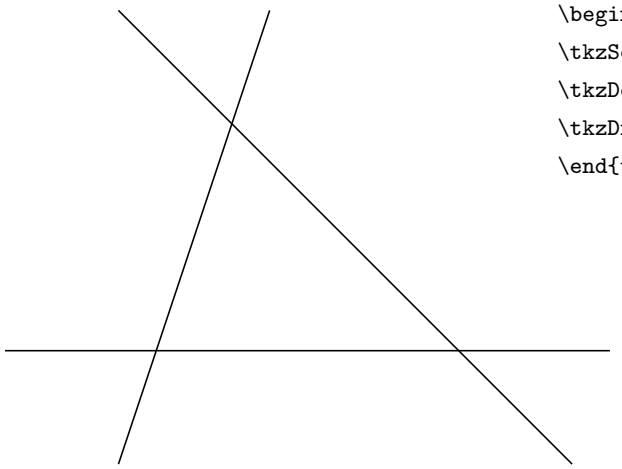

```
\begin{tikzpicture}
\tkzSetUpLine[color=blue,line width=1pt]
\begin{scope}[rotate=-90]
\tkzDefPoint(10,6){C}
\tkzDefPoint( 0,6){A}
\tkzDefPoint(10,0){B}
\tkzDefPointBy[projection = onto B-A](C)
\tkzGetPoint{H}
\tkzDrawPolygon(A,B,C)
\tkzMarkRightAngle[size=.4,fill=blue!20](B,C,A)
\tkzMarkRightAngle[size=.4,fill=red!20](B,H,C)
\tkzDrawSegment[color=red](C,H)
\end{scope}
\tkzLabelSegment[below](C,B){$a$}
\tkzLabelSegment[right](A,C){$b$}
\tkzLabelSegment[left](A,B){$c$}
\tkzLabelSegment[color=red](C,H){$h$}
\tkzDrawPoints(A,B,C)
\tkzLabelPoints[above left](H)
\tkzLabelPoints(B,C)
\tkzLabelPoints[above](A)
\end{tikzpicture}
```

21.1.2 改变线型示例




```
\begin{tikzpicture}[scale=.6]
\tkzDefPoint(1,0){A} \tkzDefPoint(4,0){B}
\tkzDefPoint(1,1){C} \tkzDefPoint(5,1){D}
\tkzDefPoint(1,2){E} \tkzDefPoint(6,2){F}
\tkzDefPoint(0,4){A'}\tkzDefPoint(3,4){B'}
\tkzCalcLength[cm](C,D)  \tkzGetLength{rCD}
\tkzCalcLength[cm](E,F)  \tkzGetLength{rEF}
\tkzInterCC[R](A',\rCD cm)(B',\rEF cm)
\tkzGetPoints{I}{J}
\tkzSetUpLine[style=dashed,color=gray]
\tkzDrawLine(A',B')
\tkzCompass(A',B')
\tkzDrawSegments(A,B C,D E,F)
\tkzDrawCircle[R](A',\rCD cm)
\tkzDrawCircle[R](B',\rEF cm)
\tkzSetUpLine[color=red]
\tkzDrawSegments(A',I B',I)
\tkzDrawPoints(A,B,C,D,E,F,A',B',I,J)
\tkzLabelPoints(A,B,C,D,E,F,A',B',I,J)
\end{tikzpicture}
```

21.1.3 线段延伸示例



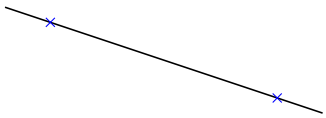
```
\begin{tikzpicture}
\tkzSetUpLine[add=.5 and .5]
\tkzDefPoints{0/0/A,4/0/B,1/3/C}
\tkzDrawLines(A,B B,C A,C)
\end{tikzpicture}
```

21.2 设置点样式

\tkzSetUpPoint[选项]

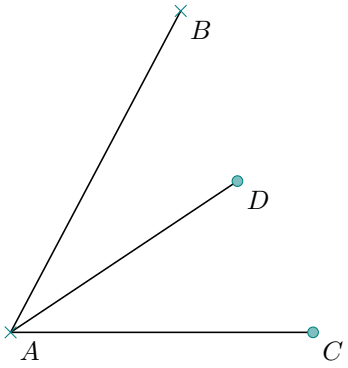
选项	默认值	含义
color	black	颜色
size	3pt	尺寸
fill	black!50	填充色
shape	circle	圆或十字线

21.2.1 示例 1



```
\begin{tikzpicture}
\tkzSetUpPoint[shape = cross out,color=blue]
\tkzInit[xmax=100,xstep=20,ymax=.5]
\tkzDefPoint(20,1){A}
\tkzDefPoint(80,0){B}
\tkzDrawLine(A,B)
\tkzDrawPoints(A,B)
\end{tikzpicture}
```

21.2.2 示例 2



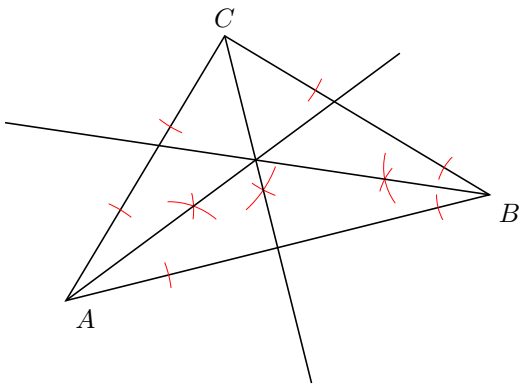

```
\begin{tikzpicture}
\tkzInit[ymin=-0.5,ymax=3,xmin=-0.5,xmax=7]
\tkzDefPoint(0,0){A}
\tkzDefPoint(02.25,04.25){B}
\tkzDefPoint(4,0){C}
\tkzDefPoint(3,2){D}
\tkzDrawSegments(A,B A,C A,D)
{\tkzSetUpPoint[shape=cross out,fill= teal!50,size=4,color=teal]
\tkzDrawPoints(A,B)}
\tkzSetUpPoint[fill= teal!50,size=4,color=teal]
\tkzDrawPoints(C,D)
\tkzLabelPoints(A,B,C,D)
\end{tikzpicture}
```

21.3 设置尺规标记样式

\tkzSetUpCompass[选项]

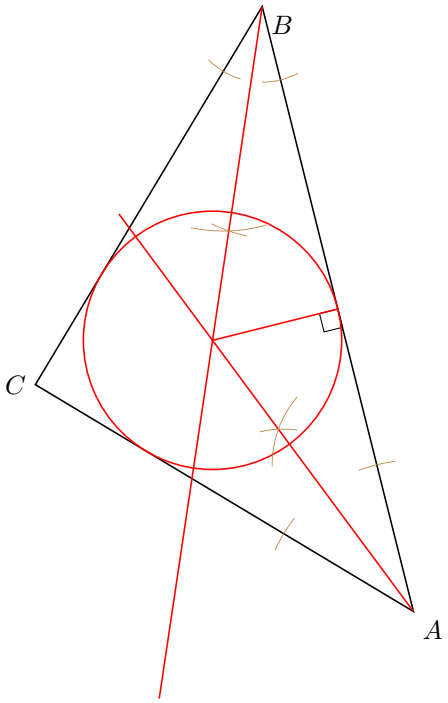
选项	默认值	含义
color	black	圆弧颜色
line width	0.4pt	圆弧线宽
style	solid	圆弧线型

21.3.1 示例 1




```
\begin{tikzpicture}[scale=0.7]
\tkzDefPoints{0/1/A, 8/3/B, 3/6/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpCompass[color=red,line width=.2 pt]
\tkzDefLine[bisector](A,C,B) \tkzGetPoint{c}
\tkzDefLine[bisector](B,A,C) \tkzGetPoint{a}
\tkzDefLine[bisector](C,B,A) \tkzGetPoint{b}
\tkzShowLine[bisector,size=2,gap=3](A,C,B)
\tkzShowLine[bisector,size=2,gap=3](B,A,C)
\tkzShowLine[bisector,size=1,gap=2](C,B,A)
\tkzDrawLines[add=0 and 0 ](B,b)
\tkzDrawLines[add=0 and -.4 ](A,a  C,c)
\tkzLabelPoints(A,B) \tkzLabelPoints[above](C)
\end{tikzpicture}
```

21.3.2 示例 2




```
\begin{tikzpicture}[scale=1,rotate=90]
\tkzDefPoints{0/1/A, 8/3/B, 3/6/C}
\tkzDrawPolygon(A,B,C)
\tkzSetUpCompass[color=brown,line width=.3 pt,style=tkzdotted]
\tkzDefLine[bisector](B,A,C)  \tkzGetPoint{a}
\tkzDefLine[bisector](C,B,A)  \tkzGetPoint{b}
\tkzInterLL(A,a)(B,b) \tkzGetPoint{I}
\tkzDefPointBy[projection= onto A-B](I)
\tkzGetPoint{H}
\tkzMarkRightAngle(I,H,A)
\tkzDrawCircle[radius,color=red](I,H)
\tkzDrawSegments[color=red](I,H)
\tkzDrawLines[add=0 and -.5,,color=red](A,a)
\tkzDrawLines[add=0 and 0,color=red](B,b)
\tkzShowLine[bisector,size=2,gap=3](B,A,C)
\tkzShowLine[bisector,size=1,gap=3](C,B,A)
\tkzLabelPoints(A,B)\tkzLabelPoints[left](C)
\end{tikzpicture}
```

21.4 局部样式设置

既可以使用 tkzSetUpPoint 设置全局样式，也使用 TikZ 选项设置局部样式。


```
\tkzSetUpPoint[color=blue!50!white, fill=gray!20!red!50!white]
\tikzset{/tikz/mystyle/.style={color=blue!20!black,fill=blue!20}}
\begin{tikzpicture}
\tkzDefPoint(0,0){O}
\tkzDefPoint(0,1){A}
\tkzDrawPoints(O) % general style
\tkzDrawPoints[mystyle,size=4](A) % my style
\tkzLabelPoints(O,A)
\end{tikzpicture}
```

第二十二章 tkz-base

22.1 tkz-base 宏包工具

首先，无需处理 TikZ 包围盒尺寸，早期的 tkz-euclide 宏包未对包围盒进行控制，现在提供了包围盒设置命令。

然而，有时也需要控制显示尺寸。

为此，需要设置工作区域包围盒，这由 tkz-base 宏包实现，该宏包提供的主要命令是 tkzInit，并建议使用 1cm 为绘图单位。某些情况下，则需要指定画布大小 (xmin、xmax、ymin 和 ymax)，并使用“裁剪”矩形尽可能控制图形尺寸。

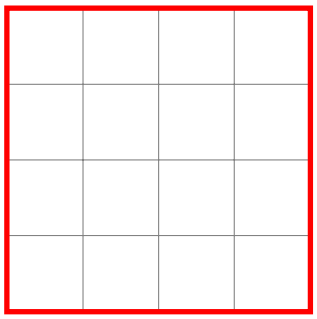
tkz-euclide 宏包使用的 tkz-base 宏包提供的两个命令是：

- \tkzInit
- \tkzClip

为实现该功能，tkz-base 宏包提供了一个命令用于操作包围盒，以查看、备份、恢复包围盒 (参见 tzk-base 宏包的 Bounding Box 小节)。

22.2 \tkzInit 命令和\tkzShowBB 命令

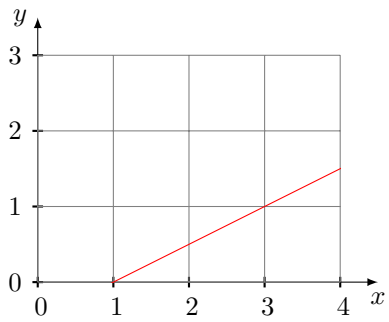
用图形四周的矩形表示包围盒。



```
\begin{tikzpicture}
\tkzInit[xmin=-1,xmax=3,ymin=-1, ymax=3]
\tkzGrid
\tkzShowBB[red,line width=2pt]
\end{tikzpicture}
```

22.3 \tkzClip 命令

通过对初始绘图矩形的“裁剪”，仅显示指定矩形范围的内容。



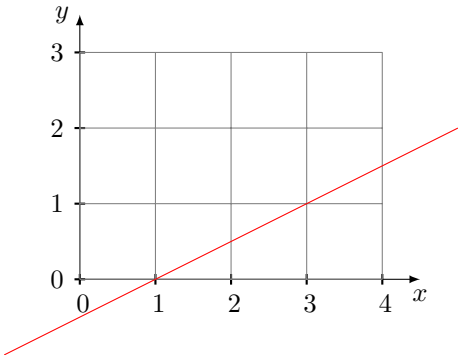

```
\begin{tikzpicture}
\tkzInit[xmax=4, ymax=3]
\tkzAxeXY
\tkzGrid
\tkzClip
\draw[red] (-1,-1)-(5,2);
\end{tikzpicture}
```

可以通过命令选项在裁剪区域四周添加指定的空间。

```
\tkzClip[space=1]
```

22.3.1 \tkzClip 命令和 space 选项示例

该选项可以裁剪区域四周添加指定的空间。




```
\begin{tikzpicture}
\tkzInit[xmax=4, ymax=3]
\tkzAxeXY
\tkzGrid
\tkzClip[space=1]
\draw[red] (-1,-1)-(5,2);
\end{tikzpicture}
```

使用 space 选项后，“ 裁剪 ” 矩形区域大小为：xmin-1、ymin-1、xmax+1 和 ymax+1。

第二十三章 常见错误

目前，根据需要，已对语法进行了多次改进，这可能会带来很多错误，主要改进有：

- 绘制多个点时，\tkzDrawPoint(A,B) 命令中需要使用\tkzDrawPoints 命令。
- 使用\tkzGetPoint(A) 得到并命名一个点时，需要使用大括号而不是小括号，因此，应该写成: \tkzGetPoint{A}
- 可以用\tkzGetPoint{A} 命令代替\tkzGetFirstPoint{A} 命令。当返回两个点时，可以使用\tkzGetPoints{A}{B} 命令一次得到两个点，也可以使用\tkzGetFirstPoint{A} 命令或\tkzGetSecondPoint{A} 命令分别得到两个点。也可以通过\tkzFirstPointResult 命令或\tkzSecondPointResult 命令引用这两个点中的一个点。
  第 3 个点可以用\tkzPointResult 命令得到。
- 需要绘制多条线段时，应该使用\tkzDrawSegments 带 “s” 的命令，而不能使用\tkzDrawSegment(A,B A,C) 命令，但注意该命令效率不高。
- 命令选项与参数需要配合使用，所有圆相关的命令都需要知道圆的半径，如果半径需要通过计算得到，则需要使用 R 命令选项。
- \tkzDrawSegments[color = gray,style=dashed]{B,B’ C,C’} 是错误语法，只有对象定义命令才使用大括号。
- 角度的度量单位常用度，极少使用弧度.
- 如果传入参数时需要计算，并发生了错误，那么建议在使用该命令前完成计算。
- 不要混合使用 pgfmath 和 xfp 的语法，本宏包主要使用 xfp 语法，但如果更喜欢 pgfmath 库的语法，则建议在传入参数前完成计算。
- 有关\tkzClip 的用法: 为了得到更为精确的结果，该宏包尽量避免向量归一化计算。向量归一化的优点是能够更好的控制对象尺寸，但其缺点是在使用 TEX 计算时，会带来精度损失。这些误差通常很小，约为千分之一，但是当图幅较大时，其累积误差可能会导致灾难。不归一化，意味着某些点可能会远离工作区域，可以使用\tkzClip 命令进行图形裁剪。
- 如果角度太小时，使用\tkzDrawAngle 命令，则会发生错误。该错误是使用在圆弧上放置标记的 decoration 装饰库而产生的，即使没有标记，该错误仍然存在。可以使用 mkpos=.2 选项避免该错误，该选项能够在绘制圆弧之前布置标记。另一种方法是使用\tkzFillAngle 命令避免该错误发生。