TiKZ-Euclide

Eureka

November 22, 2023

Contents

1 Introducing to Euclide

Hello TikZ-Euclide. TikZ-Euclide doesn't prevent you from using TikZ. It just makes your 'Euclide Drwing' easier. TikZ-Euclide is based on TikZ, which means that TiKZ can do everything that TikZ-Euclide can do.

The basic five elements of Euclide are:

`points, segments, lines, triangles, polygons, circles`

Then These proceese can be summaried as Folowing:

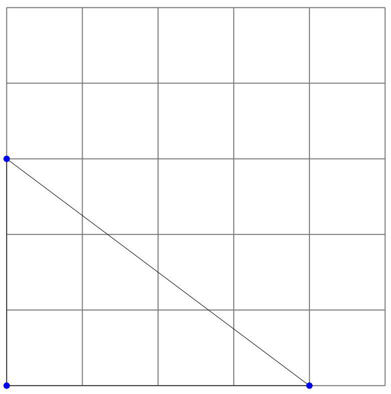
- define
- create
- draw
- mark
- label

2 An Example

2.1 Simple Drawing

To start with a Simple Example:

```
\begin{tikzpicture}
    % Init
    \tkzInit[xmax=5, ymax=5]
    \tkzGrid
    % 1. def point
    \tkzDefPoint(0, 0){A}
    \tkzDefPoint(4, 0){B}
    \tkzDefPoint(0, 3){C}
    % 2. def polygon
    \tkzDrawPolygon(A, B, C)
    % 3. show points before
    \tkzDrawPoints[color=blue](A, B, C)
\end{tikzpicture}
```



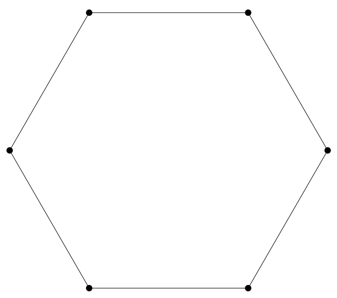
Warning:

➢ 1) `\tkzDrawPoints[color=blue] (A, B, C)` is wrong
Reason: $\longrightarrow$ `space in [] and ().`
correct: $\longrightarrow$ `\tkzDrawPoints[color=blue](A, B, C)`

➢ 2) You don't need to load `xfp, xcolor`

2.2 Loop In Euclide

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



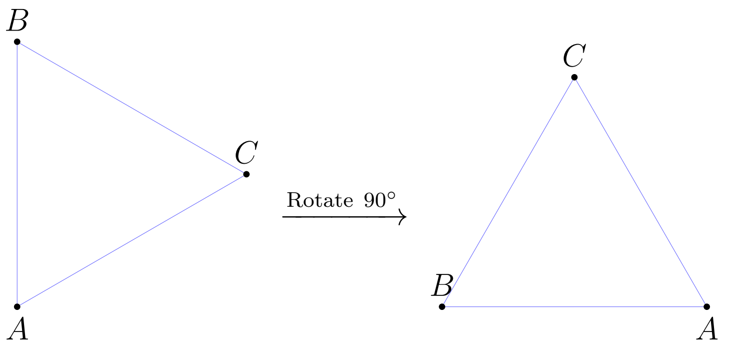
Warning: Becareful the Loop Method `(A_1, A_..., A_6)`

2.3 Reletive Point

tikz 'scope' is one way of archieve reletive point define, while TiKZ-Euclide define a command

➢ Original tikz method:

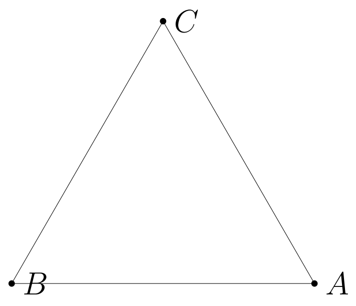
```
\tkzDefPoint(2,3){A}
\begin{scope}[shift=(A)]
    \tkzDefPoint(90:5){B}
    \tkzDefPoint(30:5){C}
\end{scope}
```



➢ TiKZ-Euclide method:

```
\tkzDefShiftPoint[A](30:3){B}
```

```
\begin{tikzpicture}
    \tkzDefPoint(0,0){A}
    \tkzDefShiftPoint[A](-4, 0){B}
    \tkzDefShiftPoint[A](120:4){C}
    \tkzDrawPolygon(A,B,C)
    \tkzDrawPoints(A,B,C)
    \tkzLabelPoints[below](A,B,C)
\end{tikzpicture}
```

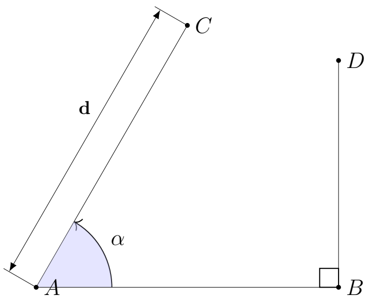


2.4 Annotate

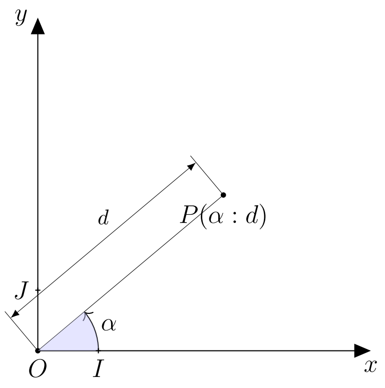
Annotate an `angle` or `Asegment` is as simple as you had ever think:

```
% 1. Draw coordinates
\tkzDrawXY[noticks,>=triangle 45]
% 2. Mark an Angle
\tkzMarkAngle[mark=none,->](I,O,P)
\tkzFillAngle[fill=blue!20, opacity=.5](I,O,P)
\tkzMarkRightAngle(I,O,P)
% 3. Annotate a segment, 'dim' is ioptional
\tkzDrawSegment[dim={$d$, <vertical distance>, above=<text vertical distance>}](O,P)
```

```
\begin{tikzpicture}[global scale=.85]
    \tkzDefPoint(0, 0){A}
    \tkzDefShiftPoint[A](4,0){B}
    \tkzDefShiftPoint[A](60:4){C}
    \tkzDrawSegment(A,B)
    \tkzDrawSegment[dim={$\mathbf{d}$,
        ↪ 1em, above=10pt}](A,C)
    \tkzMarkAngle[mark=none,->](B,A,C)
    \tkzFillAngle[fill=blue!20,
        ↪ opacity=.5](B,A,C)
    \tkzDefShiftPoint[B](90:3){D}
    \tkzMarkRightAngle(A,B,D)
    % Points Annotate
    \tkzLabelAngle[pos=1.25](B,A,C){$\alpha$}
    \tkzDrawPoints(A,B,C)
    \tkzLabelPoints[below](A,B,C)
\end{tikzpicture}
```



```
% \usepackage{tkz-base} provide \tkzDrawXY
\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}
```



3 Get Ponits

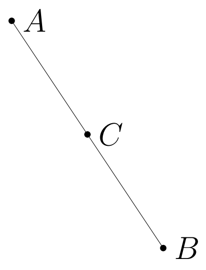
Use predifined Command, We can make get some tipycial points, such as `midpoint, center, circumcenter, orthocenter, incenter`, in a Euclide graph, Such as `segment, triangle, square, circle`.

To get the target point, we need to use:

```
\tkzGetPoints{<target point alias>}
```

just after the cmd, such as `\tkzDefMidPoint(A,B)`. There is an Example:

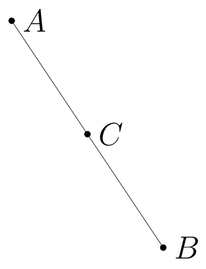
```
\begin{tikzpicture}[scale=1]
    \tkzDefPoint(2,3){A}
    \tkzDefPoint(4,0){B}
    \tkzDefMidPoint(A,B) \tkzGetPoint{C}
    \tkzDrawSegment(A,B)
    \tkzDrawPoints(A,B,C)
    \tkzLabelPoints[right](A,B,C)
\end{tikzpicture}
```



Just make it a command, and you can get the point using one command:

```
% \getmidpoint[<your target point alias>]{pt1, pt2}
\newcommand{\getmidpoint}[2][]{%
    \tkzDefMidPoint(#2) \tkzGetPoint{#1}
}
```

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



Or you can use More Advanced commnad in LaTeX3 to archieve the goal:

```
\tl_const:Nn \c_partcmd_i
% #1 = ii, iii, etc
% combine the str, then translate it to a Macro
\tl_use:c {c_partcmd_#1}
```