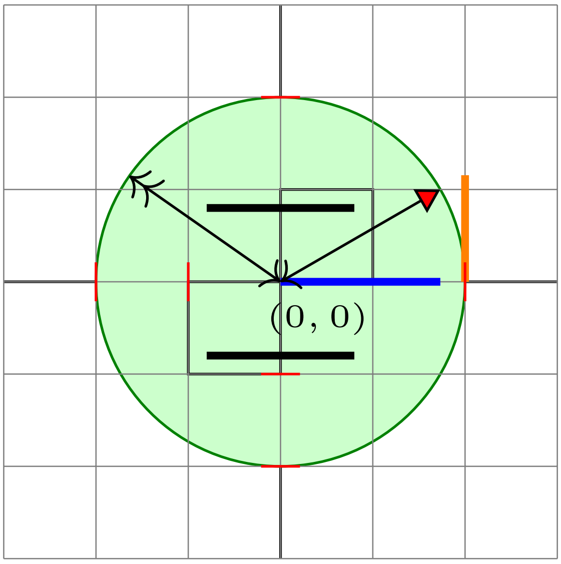
Drawing Graphs Using TikZ in Emacs Org

Eason Zhang with www.makesteamclear.com

December 13, 2019



WWW.MAKESTEAMCLEAR.COM

www.makesteamclear.com is a free project, supported by Eason Zhang, to make videos about STEAM in a more approachable and different way. If you found the contents in this post or the site or the youtube channel helpful, please consider support me, thanks

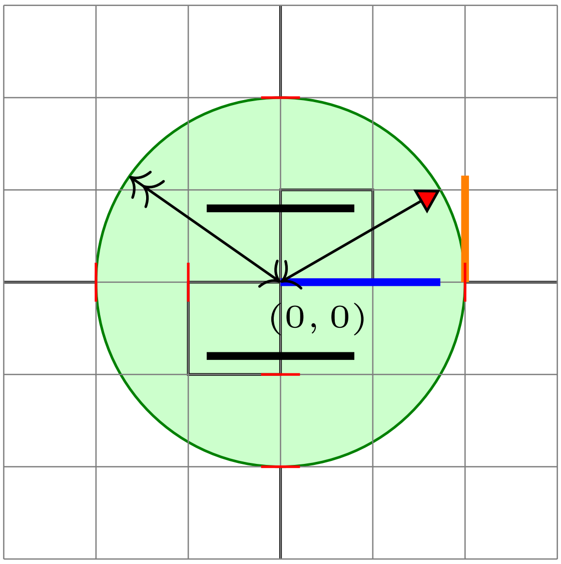
Contents

1 Drawing a TikZ picture in Emacs Org Mode

```
\usetikzlibrary{intersections,arrows.meta}
\begin{tikzpicture}[thin]
\draw (-1.5,0) -- (1.5,0);
\draw (0,-1.5) -- (0,1.5);
\filldraw[fill = green!20, draw = green!50!black] (0,0) circle[radius = 1cm];
\draw (0,0) rectangle (.5,.5);
\draw (0,0) rectangle (-0.5,-0.5);
\draw[help lines,very thin,step=.5cm,color=gray] (-1.5,-1.5) grid (1.5,1.5);
% relative coordinate
\draw[blue, very thick] (30:1) ++ (0,-0.5) --(0,0);
% name a path without drawing it
\path[name path = upward line] (1,0) -- (1,1);
\path[name path = sloped line] (0,0) -- (30:1.5cm);
% use intersection of two path
\draw[name intersections={of = upward line and sloped line, by=x}]
     [very thick, orange] (1,0) -- (x);
% use arrow
\draw[<->>] (0,0) -- (145:1);
\draw[<-{Triangle[fill=red]}] (0,0) -- (30:1);
% use scope
\begin{scope}[very thick]
\draw (-0.4,0.4) -- (0.4,0.4);
\draw (-0.4,-0.4) -- (0.4,-0.4);
\end{scope}
% use foreach
\foreach \x in {-1cm,-0.5cm,1cm}
    \draw[red] (\x,-3pt) -- (\x,3pt);
\foreach \y in {-1cm,-0.5cm,1cm}
    \draw[red](-3pt,\y) -- (3pt,\y);
% using node
\draw (0,0)+(0.2,-0.2) node {\tiny $(0,0)$ };
\end{tikzpicture}
```

Listing 1: a minimum working example

The generated figure is shown as:



1. In the minimum working example line 12 , a path is named without drawing it.
2. Line 16 gives an example of using library `intersections`. Note that you need to add the library using `\usetikzlibrary{intersections}` otherwise an error occurs during LaTeXcompiling.
3. Line 18 and 19 gives an example of using arrow. To make it work, `\usetikzlibrary{arrows.meta}` is needed. The library `arrows.meta` provides tons of types of arrows whick shock me when I see them the first time.
4. Line 21 to 24 gives an example of `scope` . In the environment, all the lines are drawn in the `very thick` style.
5. Line 26 to 29 gives an examplt of `foreach` . `foreach` is handy when you want to draw a list of objects. In the minimum working example , I draw a list of short red sticks along with the x-axis and y-axis.
6. Line 31 is an example of `node`. The keyword `node` is typically followed by some options between `[]` and then some text between `{}`. Every node has flexible anchor options to decide where the text should be placed.

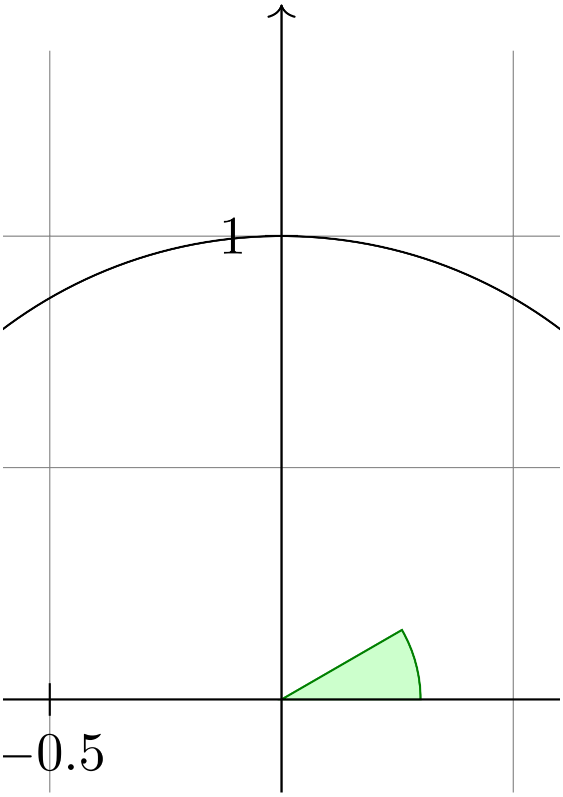
2 Another Example

The code is shown as:

```
\usetikzlibrary{intersections,arrows.meta}
\begin{tikzpicture}[scale=3]
  \clip (-0.6,-0.2) rectangle (0.6,1.51);
  \draw[step = .5cm, help lines] (-1.4,-1.4) grid (1.4,1.4);
  \filldraw[fill=green!20,draw = green!50!black] (0,0) -- (3mm,0mm)
  arc [start angle = 0, end angle = 30,radius = 3mm] -- cycle;
  \draw[->] (-1.5,0) -- (1.5,0);
  \draw[->] (0,-1.5) -- (0,1.5);
  \draw (0,0) circle [radius=1cm];
  \foreach \x in {-1,-0.5,1}
  \draw(\x cm, 1pt) -- (\x cm, -1 pt) node [anchor = north] {$\x$};
  \foreach \y in {-1,-0.5,1}
  \draw(1pt,\y cm) -- (-1pt, \y cm) node[anchor = east] {$\y$};
\end{tikzpicture}
```

Listing 2: another minimum working example

The generated figure is shown as:



3 Some Basic Rules in TikZ

1. The options appear in `[]`. No matter it is an object or an operation, the contents in the following `[]` serve as options.
   Options `[]` can be at the very beginning of the environment `tikzpicture` following the operation, following the object.
2. `\filldraw` is a good command. It draws a closed loop and fill it with color or pattern. The colors for filling and drawing can be different.
3. Coordinates can be specified in x-y format, polar format.
   - The easiest way is `(x,y)` which means `x` cm in the x-axis and `y` cm in the y-axis;
   - `(a:x)` is the polar format which means `x` cm in direction `a` degree.
4. `(<p> |- <q>)` is another way to specify coordinates for example `(30:1 |- 0,0)` which means the interaction of a vertical line through `(30:1)` and a horizontal line through `(0,0)`.

5. Relative coordinates are possible with + and ++ in front of (x,y) and (a:x) . + is relative to the closest coordinate whereas ++ is relative to the very first coordinate of current path.

4 Some tips for in TikZ

1. To use intersections to specify a coordinate, you need to include the library, i.e. \usetikzlibrary{intersections} is a must.

```
int main()
{
  int i=0;
  printf();
}
```

References