

1 Must-know commands

- tikzpicture environment
- must add ; after every command about drawing
- draw, fill, shade
- coordinate, node
- foreach

1.1 draw, fill, shade commands

Basic syntax:

`\command [parameters] (starting coordinates) arguments (argument parameters);`

command: there are some common and useful commands

draw: drawing

fill: fill the drawn region by colour

shade: fill the drawn region by colour

filldraw: show the border line colour

shadedraw: show the border line colour

parameters: **draw:** style

step= distance of each grid, default 1cm

color= colour of grid line

line width= thickness of line, accept special arguments like thin, thick, etc.

help line draw as help line style

-> add arrow head to the ending point

example: [step = 1cm, color = gray, line width = very thin]

fill: colour name, a special syntax: *color1!r!color2* represent mixed colour, *r* means the ratio of color1

shade: colour direction

- left color = color1, right color = color2
- top color = color1, right color = color2
- inner color = color1, outer color = color2

filldraw and shadedraw: specify fill/shade and draw colour

starting coordinates: starting point of drawing

arguments: what to draw

- rectangle
- grid
- parabola
- circle
- ellipse
- arc
- curve

argument parameters: depends on arguments

rectangle: diagonal vertex from starting point

```
\draw (0,0) rectangle (4,4);
```

grid: the same as rectangle

parabola: ending point

```
\draw (0,0) parabola (4,4);
```

circle: radius

```
\draw (2,2) circle (3cm);
```

ellipse: semi-major axis and semi-minor axis

```
\draw (2,2) ellipse (3cm and 1cm);
```

arc: starting angle, ending angle, radius

```
\draw (3,0) arc (0 : 75 : 3cm);
```

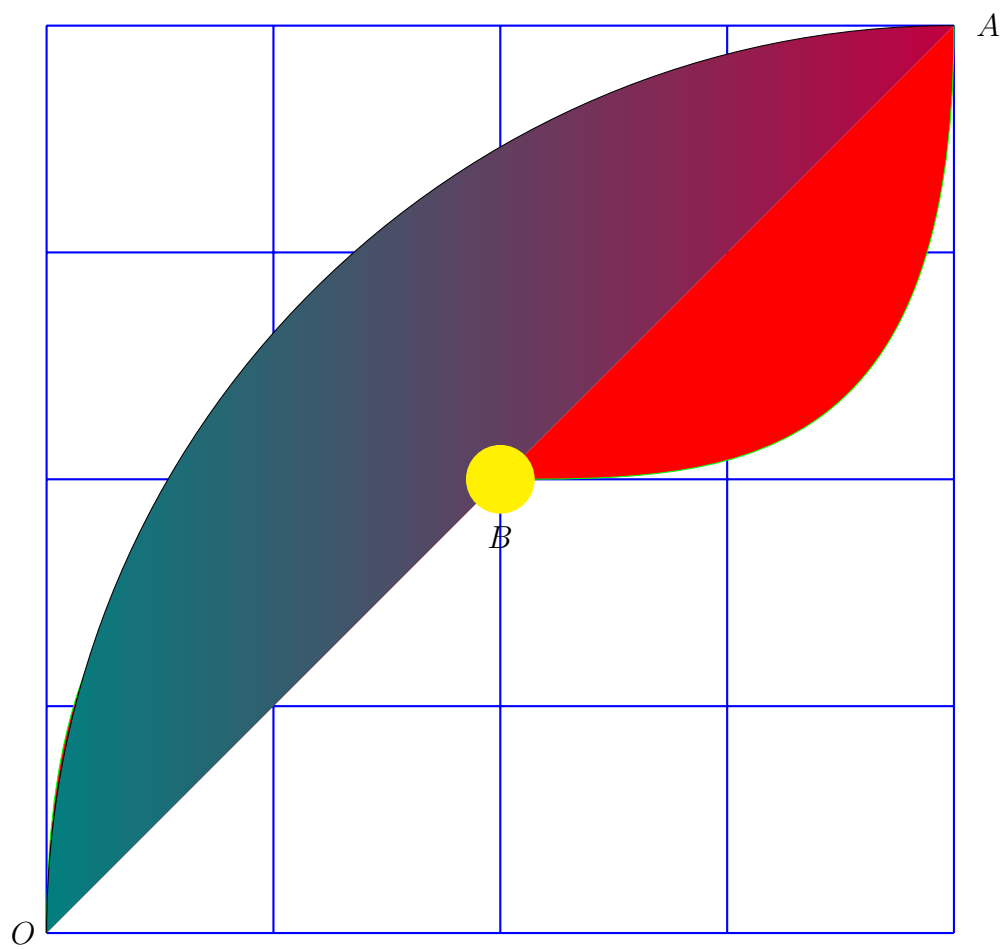
curve: .. controls "point 1" and "point2" .. ending point (1 or 2 control point)

```
\draw (0,0) .. controls (0,4) and (4,0) .. (4,4);
```

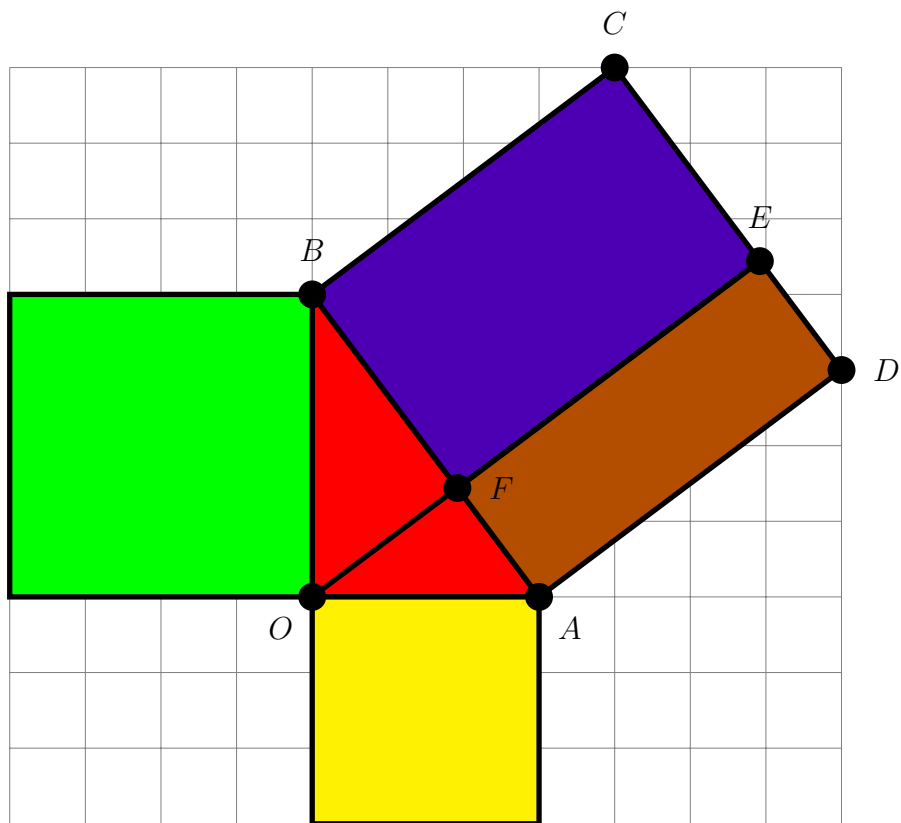
1.2 Node

Using node to point out a point and add description

2 Examples



3 Basic



4 Simple linear and parabolic function

