# decksh reference

Version

2025-11-20-1.0.0

# Introduction

decksh is a Domain-Specific Language (DSL) for making presentations, visualizations, and information displays.

This reference describes the keywords and elements of the language, how to structure decksh code, along with how to use variables, assignments, and binary operations.

Also included is a color reference and a detailed description, with examples, for all decksh elements.

# Keywords

## Structure

deck/edeck
slide/eslide
canvas
def/edef
for/efor
func
grid
import
include
if/else/eif

## Text

text
btext
ctext
etext
rtext
arctext
textblock
textblockfile
textfile
textcode

## Graphics

acircle
arc
circle
curve
ellipse
hline
line
pill
polygon
polyline
rect
rrect
ruler
square
star
vline

## Lists

list
blist
nlist
clist
li
elist

## Braces

lbrace
rbrace
ubrace
dbrace
lbracket
rbracket
dbracket
ubracket

## Arrows

arrow
rcarrow
lcarrow
ucarrow
dcarrow

## Maps

geoarc
geoborder
geoimage
geolabel
geoloc
geomark
geopath
geopathfile
georegion

## Images

image
cimage

## Charts

dchart
legend

## Assignments

polar
polarx
polary
random
area
format
substr
vmap
dump

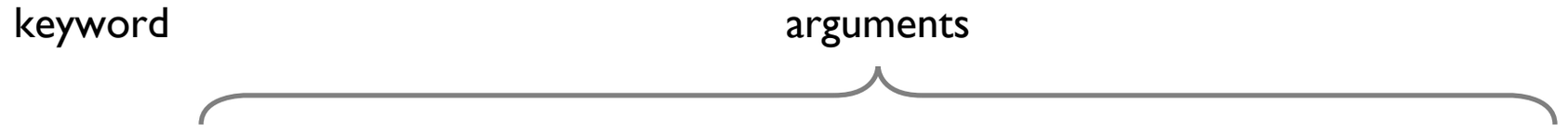
## Math

cosine
sine
sqrt
tangent

## Data

data
edata
content

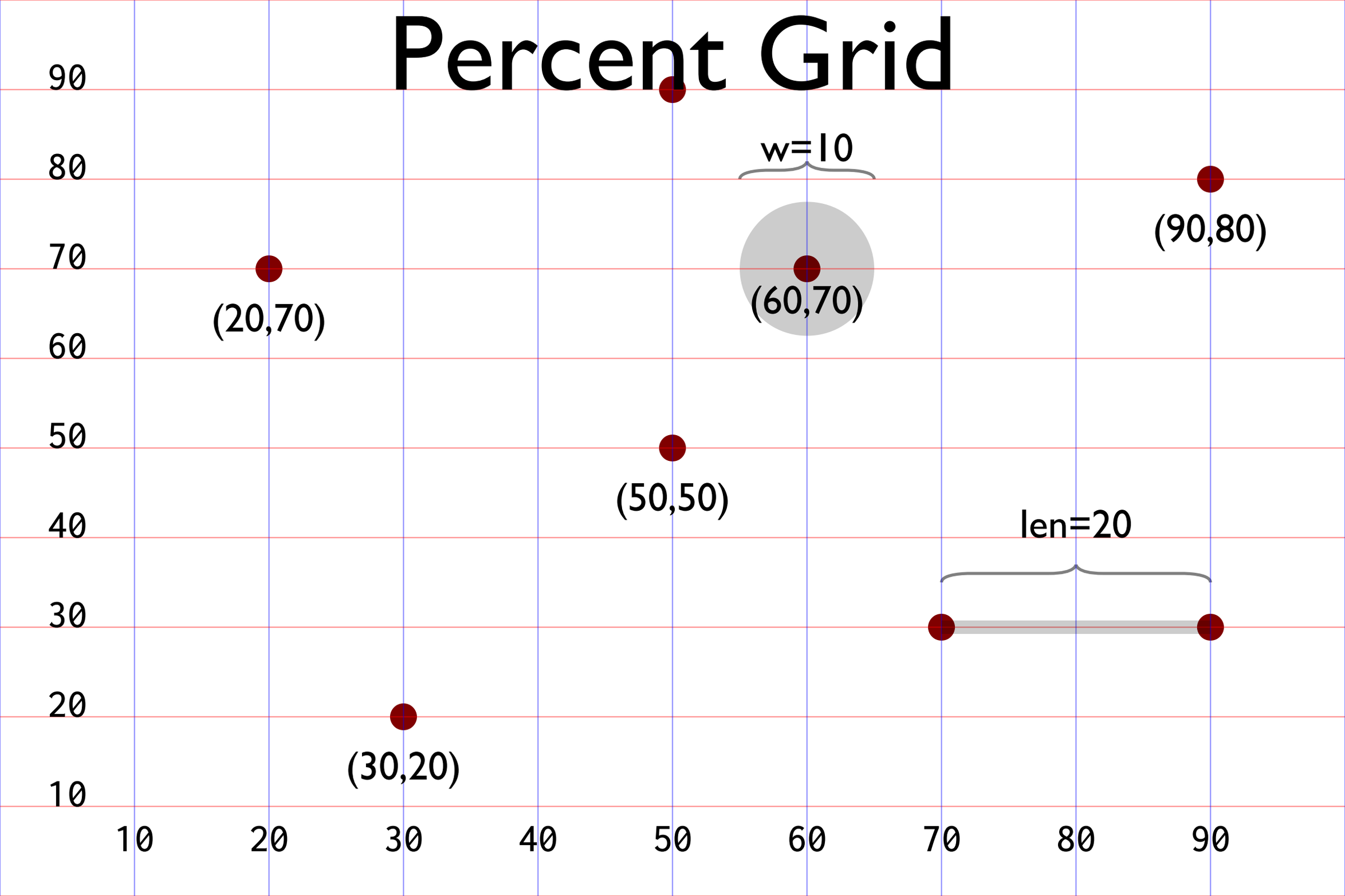
# Keywords and arguments




```
text "..string...." x  y  n "font" "color" op


text "hello, world" 80 50 2                    hello, world
text "hello, world" 80 40 2 "serif"            hello, world
text "hello, world" 80 30 2 "serif" "red"      hello, world
text "hello, world" 80 20 2 "serif" "red" 50   hello, world
```

# Structure


// This is a comment
comment
deck
canvas 1920 1080
canvas size hint (width height)
x=20 // define x
inline comment
variables
y=80
deck
slide
slide 1
text "first" x y 2
eslide
slide "black" "white"
slide 2
include "file.dsh"
eslide
edeck
ctext "hello, world" 50 25 10
circle 50 0 100 "blue"
for x=20 80 10
circle x 75 2
efor


Percent Grid
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
w=10
(20,70)
(60,70)
(90,80)
(50,50)
len=20
(30,20)

# Defining and using variables

name is thing

```
x = 3.14159265
```
number

```
s = "hello"
```
string

```
y = x
```
another variable

```
x = a + b
```
binary operation

```
x *= 10
```
assignment operator

```
text s x y 2
```
variable use

```
dump [var list]
```
dump the value of variables

Note: the reserved variable deckshVersion contains the version of decksh `2025-11-20-1.0.0`

# Binary operators

| | |
|---|---|
| `x = a + b` | addition |
| `x = a - b` | subtraction |
| `x = a * b` | multiplication |
| `x = a / b` | division |
| `x = a % b` | modulo |

# Assignment operators

| | |
|---|---|
| `x += 10` | increase x by 10 |
| `x -= 10` | decrease x by 10 |
| `x *= 10` | multiply x by 10 |
| `x /= 10` | divide x by 10 |

# Special Assignments

| | |
|---|---|
| `p=(expr, expr)` | coordinates (p_x, p_y) |
| `p=polar cx cy r theta` | polar coordinate (p_x, p_y) |
| `x=polarx cx cy r theta` | polar coordinate (x) |
| `y=polary cx cy r theta` | polar coordinate (y) |
| `v=format string expr` | number formatting |
| `v=substr string begin end` | substring |
| `v=random v1 v2` | random number |
| `v=vmap data v1 v2 v3 v4` | range map |
| `v=area expr` | area |
| `v=cosine expr` | cosine |
| `v=sine expr` | sine |
| `v=sqrt expr` | square root |
| `v=tangent expr` | tangent |

# Colors, fonts, opacity, gradient

| Colors | Fonts | | Opacity (0-100) |
|---|---|---|---|
| `"steelblue"` | `"sans"` | Sans Serif | 100  |
| `"#4682b4"` | `"serif"` | Serif | 50  |
| `"rgb(70,130,180)"` | `"mono"` | Monospace | 25  |
| `"hsv(207,61,71)"` | `"symbol"` | ❀✲❀❄❄ | 10  |

`"red/blue/90"` 

(applies to rect and square only)

# Color Index

| name | hex | RGB |
|---|---|---|
| aliceblue | #f0f8ff | rgb(240,248,255) |
| antiquewhite | #faebd7 | rgb(250,235,215) |
| aqua | #00ffff | rgb(0,255,255) |
| aquamarine | #7fffd4 | rgb(127,255,212) |
| azure | #f0ffff | rgb(240,255,255) |
| beige | #f5f5dc | rgb(245,245,220) |
| bisque | #ffe4c4 | rgb(255,228,196) |
| black | #000000 | rgb(0,0,0) |
| blanchedalmond | #ffebcd | rgb(255,235,205) |
| blue | #0000ff | rgb(0,0,255) |
| blueviolet | #8a2be2 | rgb(138,43,226) |
| brown | #a52a2a | rgb(165,42,42) |
| burlywood | #deb887 | rgb(222,184,135) |
| cadetblue | #5f9ea0 | rgb(95,158,160) |
| chartreuse | #7fff00 | rgb(127,255,0) |
| chocolate | #d2691e | rgb(210,105,30) |

| name | hex | RGB |
|---|---|---|
| coral | #ff7f50 | rgb(255,127,80) |
| cornflowerblue | #6495ed | rgb(100,149,237) |
| cornsilk | #fff8dc | rgb(255,248,220) |
| crimson | #dc143c | rgb(220,20,60) |
| cyan | #00ffff | rgb(0,255,255) |
| darkblue | #00008b | rgb(0,0,139) |
| darkcyan | #008b8b | rgb(0,139,139) |
| darkgoldenrod | #b8860b | rgb(184,134,11) |
| darkgray | #a9a9a9 | rgb(169,169,169) |
| darkgreen | #006400 | rgb(0,100,0) |
| darkgrey | #a9a9a9 | rgb(169,169,169) |
| darkkhaki | #bdb76b | rgb(189,183,107) |
| darkmagenta | #8b008b | rgb(139,0,139) |
| darkolivegreen | #556b2f | rgb(85,107,47) |
| darkorange | #ff8c00 | rgb(255,140,0) |
| darkorchid | #9932cc | rgb(153,50,204) |

# Color Index (2)

| name | hex | RGB |
|---|---|---|
| darkred | #8b0000 | rgb(139,0,0) |
| darksalmon | #e9967a | rgb(233,150,122) |
| darkseagreen | #8fbc8f | rgb(143,188,143) |
| darkslateblue | #483d8b | rgb(72,61,139) |
| darkslategray | #2f4f4f | rgb(47,79,79) |
| darkslategrey | #2f4f4f | rgb(47,79,79) |
| darkturquoise | #00ced1 | rgb(0,206,209) |
| darkviolet | #9400d3 | rgb(148,0,211) |
| deeppink | #ff1493 | rgb(255,20,147) |
| deepskyblue | #00bfff | rgb(0,191,255) |
| dimgray | #696969 | rgb(105,105,105) |
| dimgrey | #696969 | rgb(105,105,105) |
| dodgerblue | #1e90ff | rgb(30,144,255) |
| firebrick | #b22222 | rgb(178,34,34) |
| floralwhite | #fffaf0 | rgb(255,250,240) |
| forestgreen | #228b22 | rgb(34,139,34) |

| name | hex | RGB |
|---|---|---|
| fuchsia | #ff00ff | rgb(255,0,255) |
| gainsboro | #dcdcdc | rgb(220,220,220) |
| ghostwhite | #f8f8ff | rgb(248,248,255) |
| gold | #ffd700 | rgb(255,215,0) |
| goldenrod | #daa520 | rgb(218,165,32) |
| gray | #808080 | rgb(128,128,128) |
| green | #008000 | rgb(0,128,0) |
| greenyellow | #adff2f | rgb(173,255,47) |
| grey | #808080 | rgb(128,128,128) |
| honeydew | #f0fff0 | rgb(240,255,240) |
| hotpink | #ff69b4 | rgb(255,105,180) |
| indianred | #cd5c5c | rgb(205,92,92) |
| indigo | #4b0082 | rgb(75,0,130) |
| ivory | #fffff0 | rgb(255,255,240) |
| khaki | #f0e68c | rgb(240,230,140) |
| lavender | #e6e6fa | rgb(230,230,250) |

# Color Index (3)

| name | hex | RGB |
|---|---|---|
| lavenderblush | #fff0f5 | rgb(255,240,245) |
| lawngreen | #7cfc00 | rgb(124,252,0) |
| lemonchiffon | #fffacd | rgb(255,250,205) |
| lightblue | #add8e6 | rgb(173,216,230) |
| lightcoral | #f08080 | rgb(240,128,128) |
| lightcyan | #e0ffff | rgb(224,255,255) |
| lightgoldenrodyellow | #fafad2 | rgb(250,250,210) |
| lightgray | #d3d3d3 | rgb(211,211,211) |
| lightgreen | #90ee90 | rgb(144,238,144) |
| lightgrey | #d3d3d3 | rgb(211,211,211) |
| lightpink | #ffb6c1 | rgb(255,182,193) |
| lightsalmon | #ffa07a | rgb(255,160,122) |
| lightseagreen | #20b2aa | rgb(32,178,170) |
| lightskyblue | #87cefa | rgb(135,206,250) |
| lightslategray | #778899 | rgb(119,136,153) |
| lightslategrey | #778899 | rgb(119,136,153) |

| name | hex | RGB |
|---|---|---|
| lightsteelblue | #b0c4de | rgb(176,196,222) |
| lightyellow | #ffffe0 | rgb(255,255,224) |
| lime | #00ff00 | rgb(0,255,0) |
| limegreen | #32cd32 | rgb(50,205,50) |
| linen | #faf0e6 | rgb(250,240,230) |
| magenta | #ff00ff | rgb(255,0,255) |
| maroon | #800000 | rgb(128,0,0) |
| mediumaquamarine | #66cdaa | rgb(102,205,170) |
| mediumblue | #0000cd | rgb(0,0,205) |
| mediumorchid | #ba55d3 | rgb(186,85,211) |
| mediumpurple | #9370db | rgb(147,112,219) |
| mediumseagreen | #3cb371 | rgb(60,179,113) |
| mediumslateblue | #7b68ee | rgb(123,104,238) |
| mediumspringgreen | #00fa9a | rgb(0,250,154) |
| mediumturquoise | #48d1cc | rgb(72,209,204) |
| mediumvioletred | #c71585 | rgb(199,21,133) |

# Color Index (4)

| name | hex | RGB |
|---|---|---|
| midnightblue | #191970 | rgb(25,25,112) |
| mintcream | #f5fffa | rgb(245,255,250) |
| mistyrose | #ffe4e1 | rgb(255,228,225) |
| moccasin | #ffe4b5 | rgb(255,228,181) |
| navajowhite | #ffdead | rgb(255,222,173) |
| navy | #000080 | rgb(0,0,128) |
| oldlace | #fdf5e6 | rgb(253,245,230) |
| olive | #808000 | rgb(128,128,0) |
| olivedrab | #6b8e23 | rgb(107,142,35) |
| orange | #ffa500 | rgb(255,165,0) |
| orangered | #ff4500 | rgb(255,69,0) |
| orchid | #da70d6 | rgb(218,112,214) |
| palegoldenrod | #eee8aa | rgb(238,232,170) |
| palegreen | #98fb98 | rgb(152,251,152) |
| paleturquoise | #afeeee | rgb(175,238,238) |
| palevioletred | #db7093 | rgb(219,112,147) |

| name | hex | RGB |
|---|---|---|
| papayawhip | #ffefd5 | rgb(255,239,213) |
| peachpuff | #ffdab9 | rgb(255,218,185) |
| peru | #cd853f | rgb(205,133,63) |
| pink | #ffc0cb | rgb(255,192,203) |
| plum | #dda0dd | rgb(221,160,221) |
| powderblue | #b0e0e6 | rgb(176,224,230) |
| purple | #800080 | rgb(128,0,128) |
| red | #ff0000 | rgb(255,0,0) |
| rosybrown | #bc8f8f | rgb(188,143,143) |
| royalblue | #4169e1 | rgb(65,105,225) |
| saddlebrown | #8b4513 | rgb(139,69,19) |
| salmon | #fa8072 | rgb(250,128,114) |
| sandybrown | #f4a460 | rgb(244,164,96) |
| seagreen | #2e8b57 | rgb(46,139,87) |
| seashell | #fff5ee | rgb(255,245,238) |
| sienna | #a0522d | rgb(160,82,45) |

# Color Index (5)

| name | hex | RGB |
|---|---|---|
| silver | #c0c0c0 | rgb(192,192,192) |
| skyblue | #87ceeb | rgb(135,206,235) |
| slateblue | #6a5acd | rgb(106,90,205) |
| slategray | #708090 | rgb(112,128,144) |
| slategrey | #708090 | rgb(112,128,144) |
| snow | #fffafa | rgb(255,250,250) |
| springgreen | #00ff7f | rgb(0,255,127) |
| steelblue | #4682b4 | rgb(70,130,180) |
| tan | #d2b48c | rgb(210,180,140) |
| teal | #008080 | rgb(0,128,128) |
| thistle | #d8bfd8 | rgb(216,191,216) |
| tomato | #ff6347 | rgb(255,99,71) |
| turquoise | #40e0d0 | rgb(64,224,208) |
| violet | #ee82ee | rgb(238,130,238) |
| wheat | #f5deb3 | rgb(245,222,179) |
| white | #ffffff | rgb(255,255,255) |
| whitesmoke | #f5f5f5 | rgb(245,245,245) |
| yellow | #ffff00 | rgb(255,255,0) |
| yellowgreen | #9acd32 | rgb(154,205,50) |

# Neutrals

| name | hex | RGB |
|---|---|---|
| aliceblue | #f0f8ff | rgb(240,248,255) |
| antiquewhite | #faebd7 | rgb(250,235,215) |
| azure | #f0ffff | rgb(240,255,255) |
| beige | #f5f5dc | rgb(245,245,220) |
| bisque | #ffe4c4 | rgb(255,228,196) |
| black | #000000 | rgb(0,0,0) |
| blanchedalmond | #ffebcd | rgb(255,235,205) |
| brown | #a52a2a | rgb(165,42,42) |
| burlywood | #deb887 | rgb(222,184,135) |
| chocolate | #d2691e | rgb(210,105,30) |
| cornsilk | #fff8dc | rgb(255,248,220) |
| darkgray | #a9a9a9 | rgb(169,169,169) |
| darkgrey | #a9a9a9 | rgb(169,169,169) |
| darksalmon | #e9967a | rgb(233,150,122) |
| darkslategray | #2f4f4f | rgb(47,79,79) |
| darkslategrey | #2f4f4f | rgb(47,79,79) |

| name | hex | RGB |
|---|---|---|
| dimgray | #696969 | rgb(105,105,105) |
| dimgrey | #696969 | rgb(105,105,105) |
| floralwhite | #fffaf0 | rgb(255,250,240) |
| gainsboro | #dcdcdc | rgb(220,220,220) |
| ghostwhite | #f8f8ff | rgb(248,248,255) |
| gray | #808080 | rgb(128,128,128) |
| grey | #808080 | rgb(128,128,128) |
| honeydew | #f0fff0 | rgb(240,255,240) |
| ivory | #fffff0 | rgb(255,255,240) |
| lavender | #e6e6fa | rgb(230,230,250) |
| lavenderblush | #fff0f5 | rgb(255,240,245) |
| lightgray | #d3d3d3 | rgb(211,211,211) |
| lightgrey | #d3d3d3 | rgb(211,211,211) |
| lightslategray | #778899 | rgb(119,136,153) |
| lightslategrey | #778899 | rgb(119,136,153) |
| linen | #faf0e6 | rgb(250,240,230) |

# Neutrals (2)

| name | hex | RGB |
|---|---|---|
| mintcream | #f5fffa | rgb(245,255,250) |
| mistyrose | #ffe4e1 | rgb(255,228,225) |
| moccasin | #ffe4b5 | rgb(255,228,181) |
| navajowhite | #ffdead | rgb(255,222,173) |
| oldlace | #fdf5e6 | rgb(253,245,230) |
| papayawhip | #ffefd5 | rgb(255,239,213) |
| peachpuff | #ffdab9 | rgb(255,218,185) |
| peru | #cd853f | rgb(205,133,63) |
| rosybrown | #bc8f8f | rgb(188,143,143) |
| saddlebrown | #8b4513 | rgb(139,69,19) |
| salmon | #fa8072 | rgb(250,128,114) |
| sandybrown | #f4a460 | rgb(244,164,96) |
| seashell | #fff5ee | rgb(255,245,238) |
| sienna | #a0522d | rgb(160,82,45) |
| silver | #c0c0c0 | rgb(192,192,192) |
| slategray | #708090 | rgb(112,128,144) |

| name | hex | RGB |
|---|---|---|
| slategrey | #708090 | rgb(112,128,144) |
| snow | #fffafa | rgb(255,250,250) |
| tan | #d2b48c | rgb(210,180,140) |
| wheat | #f5deb3 | rgb(245,222,179) |
| white | #ffffff | rgb(255,255,255) |
| whitesmoke | #f5f5f5 | rgb(245,245,245) |

# Reds

| name | hex | RGB |
|---|---|---|
| coral | #ff7f50 | rgb(255,127,80) |
| crimson | #dc143c | rgb(220,20,60) |
| darkmagenta | #8b008b | rgb(139,0,139) |
| darkred | #8b0000 | rgb(139,0,0) |
| deeppink | #ff1493 | rgb(255,20,147) |
| firebrick | #b22222 | rgb(178,34,34) |
| fuchsia | #ff00ff | rgb(255,0,255) |
| hotpink | #ff69b4 | rgb(255,105,180) |
| indianred | #cd5c5c | rgb(205,92,92) |
| lightcoral | #f08080 | rgb(240,128,128) |
| lightpink | #ffb6c1 | rgb(255,182,193) |
| lightsalmon | #ffa07a | rgb(255,160,122) |
| magenta | #ff00ff | rgb(255,0,255) |
| maroon | #800000 | rgb(128,0,0) |
| orangered | #ff4500 | rgb(255,69,0) |
| orchid | #da70d6 | rgb(218,112,214) |

| name | hex | RGB |
|---|---|---|
| palevioletred | #db7093 | rgb(219,112,147) |
| pink | #ffc0cb | rgb(255,192,203) |
| plum | #dda0dd | rgb(221,160,221) |
| red | #ff0000 | rgb(255,0,0) |
| thistle | #d8bfd8 | rgb(216,191,216) |
| tomato | #ff6347 | rgb(255,99,71) |

# Greens

| name | hex | RGB |
|---|---|---|
| aquamarine | #7fffd4 | rgb(127,255,212) |
| chartreuse | #7fff00 | rgb(127,255,0) |
| darkgreen | #006400 | rgb(0,100,0) |
| darkkhaki | #bdb76b | rgb(189,183,107) |
| darkolivegreen | #556b2f | rgb(85,107,47) |
| darkseagreen | #8fbc8f | rgb(143,188,143) |
| forestgreen | #228b22 | rgb(34,139,34) |
| green | #008000 | rgb(0,128,0) |
| greenyellow | #adff2f | rgb(173,255,47) |
| lawngreen | #7cfc00 | rgb(124,252,0) |
| lightgreen | #90ee90 | rgb(144,238,144) |
| lightseagreen | #20b2aa | rgb(32,178,170) |
| lime | #00ff00 | rgb(0,255,0) |
| limegreen | #32cd32 | rgb(50,205,50) |
| mediumseagreen | #3cb371 | rgb(60,179,113) |
| mediumspringgreen | #00fa9a | rgb(0,250,154) |

| name | hex | RGB |
|---|---|---|
| olive | #808000 | rgb(128,128,0) |
| olivedrab | #6b8e23 | rgb(107,142,35) |
| palegreen | #98fb98 | rgb(152,251,152) |
| seagreen | #2e8b57 | rgb(46,139,87) |
| springgreen | #00ff7f | rgb(0,255,127) |
| teal | #008080 | rgb(0,128,128) |
| yellowgreen | #9acd32 | rgb(154,205,50) |

# Blues

| name | hex | RGB |
|---|---|---|
| aqua | #00ffff | rgb(0,255,255) |
| blue | #0000ff | rgb(0,0,255) |
| cadetblue | #5f9ea0 | rgb(95,158,160) |
| cornflowerblue | #6495ed | rgb(100,149,237) |
| cyan | #00ffff | rgb(0,255,255) |
| darkblue | #00008b | rgb(0,0,139) |
| darkcyan | #008b8b | rgb(0,139,139) |
| darkslateblue | #483d8b | rgb(72,61,139) |
| darkturquoise | #00ced1 | rgb(0,206,209) |
| deepskyblue | #00bfff | rgb(0,191,255) |
| dodgerblue | #1e90ff | rgb(30,144,255) |
| lightblue | #add8e6 | rgb(173,216,230) |
| lightcyan | #e0ffff | rgb(224,255,255) |
| lightskyblue | #87cefa | rgb(135,206,250) |
| lightsteelblue | #b0c4de | rgb(176,196,222) |
| mediumaquamarine | #66cdaa | rgb(102,205,170) |

| name | hex | RGB |
|---|---|---|
| mediumblue | #0000cd | rgb(0,0,205) |
| mediumslateblue | #7b68ee | rgb(123,104,238) |
| mediumturquoise | #48d1cc | rgb(72,209,204) |
| midnightblue | #191970 | rgb(25,25,112) |
| navy | #000080 | rgb(0,0,128) |
| paleturquoise | #afeeee | rgb(175,238,238) |
| powderblue | #b0e0e6 | rgb(176,224,230) |
| royalblue | #4169e1 | rgb(65,105,225) |
| skyblue | #87ceeb | rgb(135,206,235) |
| slateblue | #6a5acd | rgb(106,90,205) |
| steelblue | #4682b4 | rgb(70,130,180) |
| turquoise | #40e0d0 | rgb(64,224,208) |

# Violets

| name | hex | RGB |
| --- | --- | --- |
| blueviolet | #8a2be2 | rgb(138,43,226) |
| darkorchid | #9932cc | rgb(153,50,204) |
| darkviolet | #9400d3 | rgb(148,0,211) |
| indigo | #4b0082 | rgb(75,0,130) |
| mediumorchid | #ba55d3 | rgb(186,85,211) |
| mediumpurple | #9370db | rgb(147,112,219) |
| mediumvioletred | #c71585 | rgb(199,21,133) |
| purple | #800080 | rgb(128,0,128) |
| violet | #ee82ee | rgb(238,130,238) |

# Yellows

| name | hex | RGB |
|---|---|---|
| darkgoldenrod | #b8860b | rgb(184,134,11) |
| darkorange | #ff8c00 | rgb(255,140,0) |
| gold | #ffd700 | rgb(255,215,0) |
| goldenrod | #daa520 | rgb(218,165,32) |
| khaki | #f0e68c | rgb(240,230,140) |
| lemonchiffon | #fffacd | rgb(255,250,205) |
| lightgoldenrodyellow | #fafad2 | rgb(250,250,210) |
| lightyellow | #ffffe0 | rgb(255,255,224) |
| orange | #ffa500 | rgb(255,165,0) |
| palegoldenrod | #eee8aa | rgb(238,232,170) |
| yellow | #ffff00 | rgb(255,255,0) |

(b)text ctext etext rtext arctext textbox textfile textcode line

begin center end

rotate

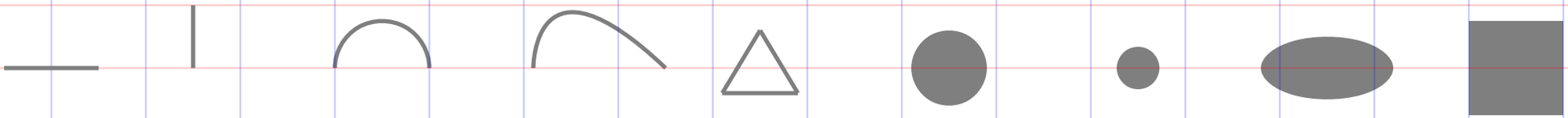
hello, world

Now is the time
for all good men
to come

This is the contents
of a file. it has lines of text.
Reading is fundamental.

```
import "fmt"
func main() {
  fmt.Println("Go")
}
```

hline vline arc curve polyline circle acircle ellipse square

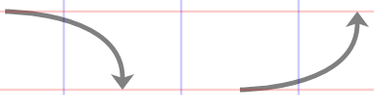


rect rrect pill polygon star image cimage lbrace rbrace

sky

ubrace dbrace lbracket rbracket ubracket dbracket arrow lcarrow rcarrow

dcarrow ucarrow list blist nlist clist



one
two
three

- one
- two
- three

1. one
2. two
3. three

first
second item
thrid

decksh version
2025-11-20-1.0.0

# Textual Elements

| description | keyword | mandatory | optional |
|---|---|---|---|
| Left-aligned | `text` | `"..." x y fontsize` | `font color op link` |
| Centered | `ctext` | `"..." x y fontsize` | `font color op link` |
| End-aligned | `etext` | `"..." x y fontsize` | `font color op link` |
| Rotated | `rtext` | `"..." x y angle fontsize` | `font color op link` |
| Text on an arc | `arctext` | `"..." x y rad a1 a2 fontsize` | `font color op link` |
| Block text | `textblock` | `"..." x y w fontsize` | `font color op link` |
| Block text from file | `textblockfile` | `"file" x y w fontsize` | `font color op link` |
| File contents | `textfile` | `"file" x y fontsize` | `font color op spacing` |
| Code listing | `textcode` | `"file" x y w fontsize` | `font color` |

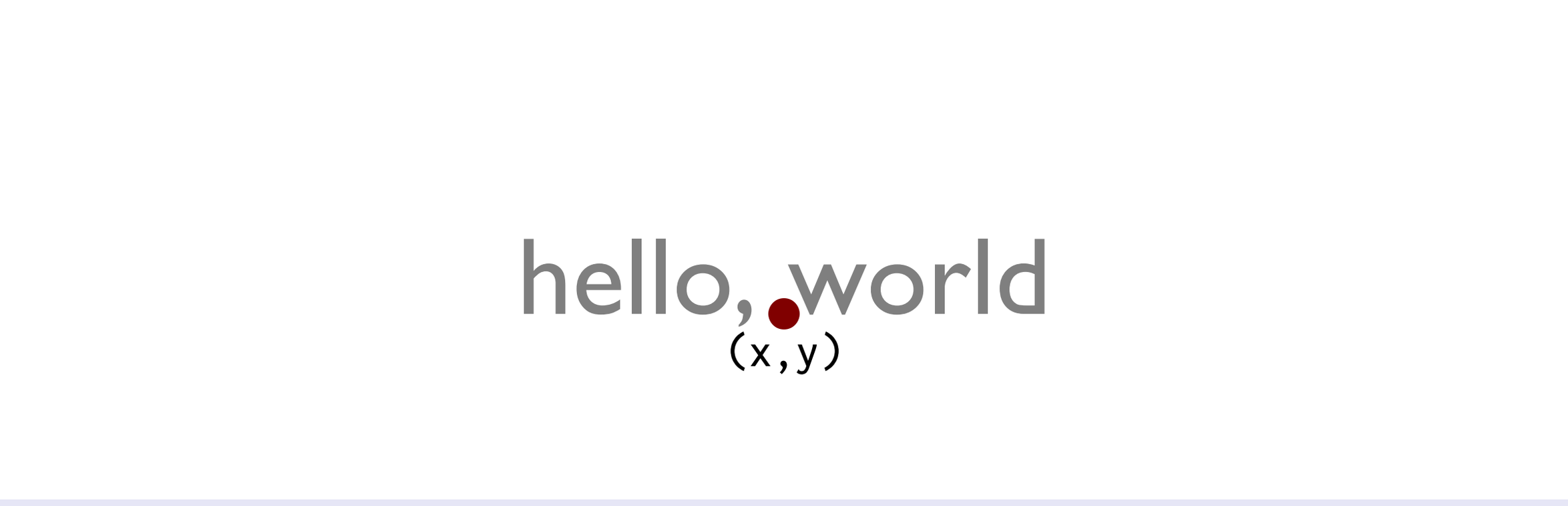

```
text "..." x y fontsize font color op link
```

abc

```
text "abc" 20 20 4
```

abc

```
text "abc" 75 20 7 "mono" "maroon"
```



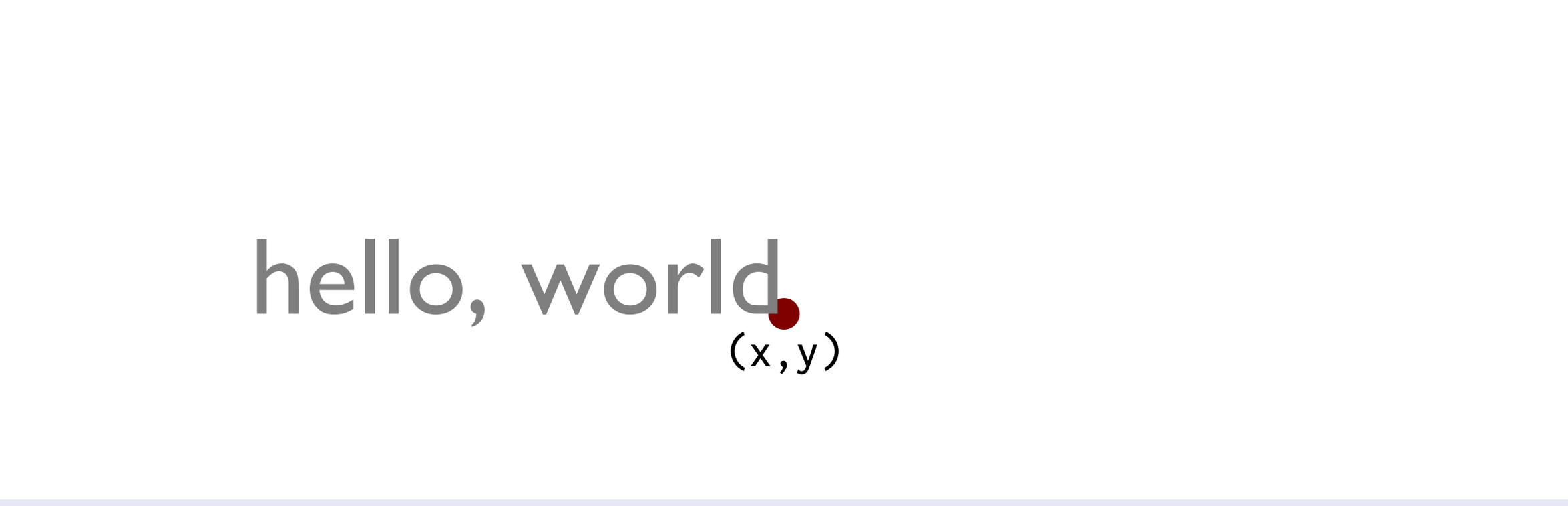

```
ctext "..." x y fontsize  font color op link
```

abc

```
ctext "abc" 20 20 4
```

abc

```
ctext "abc" 80 20 7 "mono" "maroon"
```




```
etext "..." x y fontsize  font color op link
```

abc

```
etext "abc" 20 20 4
```

abc

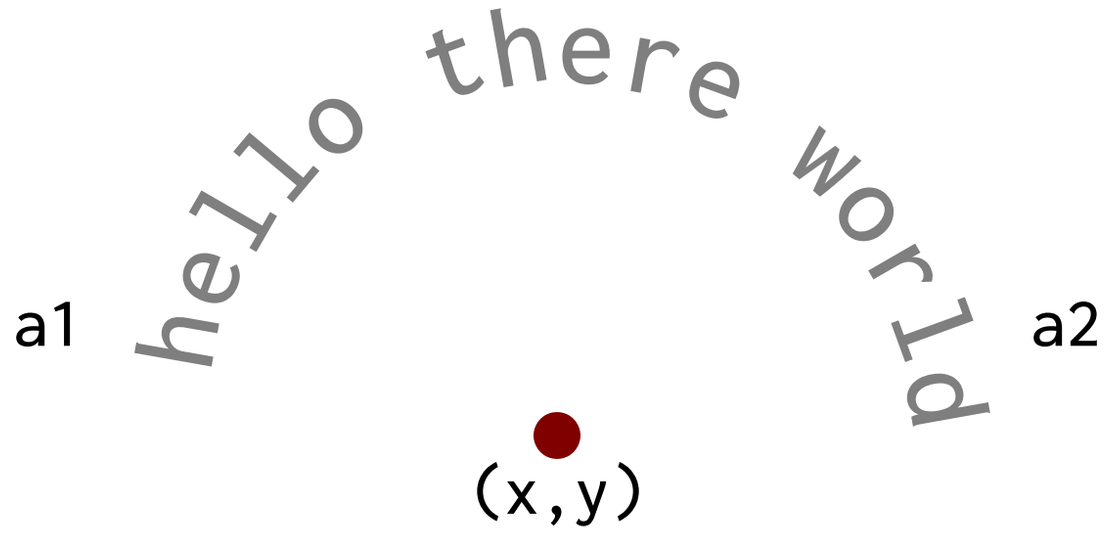
```
etext "abc" 80 20 7 "mono" "maroon"
```


```
rtext "..." x y angle fontsize  font color op link
```


```
rtext 20 20 30 3
```

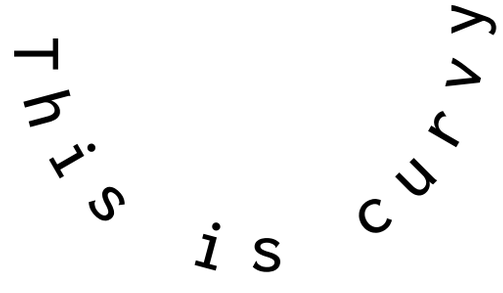

```
rtext 50 20 90 5
```


```
rtext 80 20 270 4 "sans" "maroon"
```

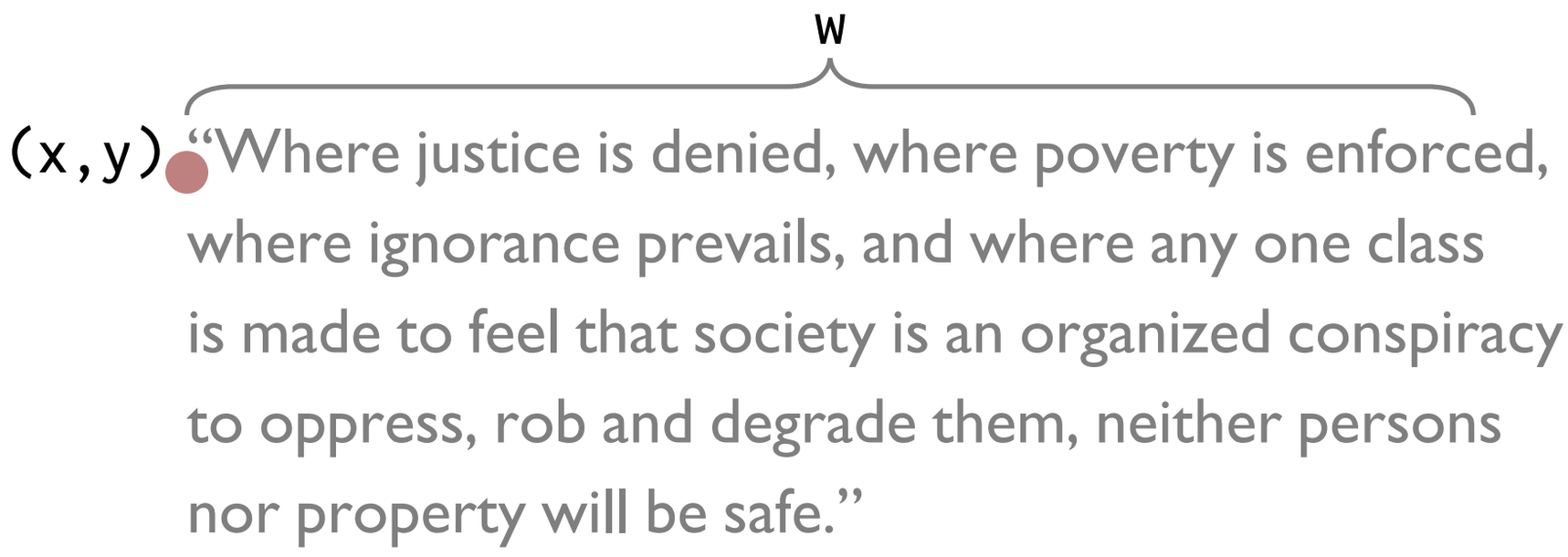



```
arctext "..." x y radius a1 a2 fontsize font color op link
```


```
arctext "What is up" 25 20 10 180 90 3 "mono"
```




```
arctext "This is curvy" 75 30 10 180 360 3 "mono"
```



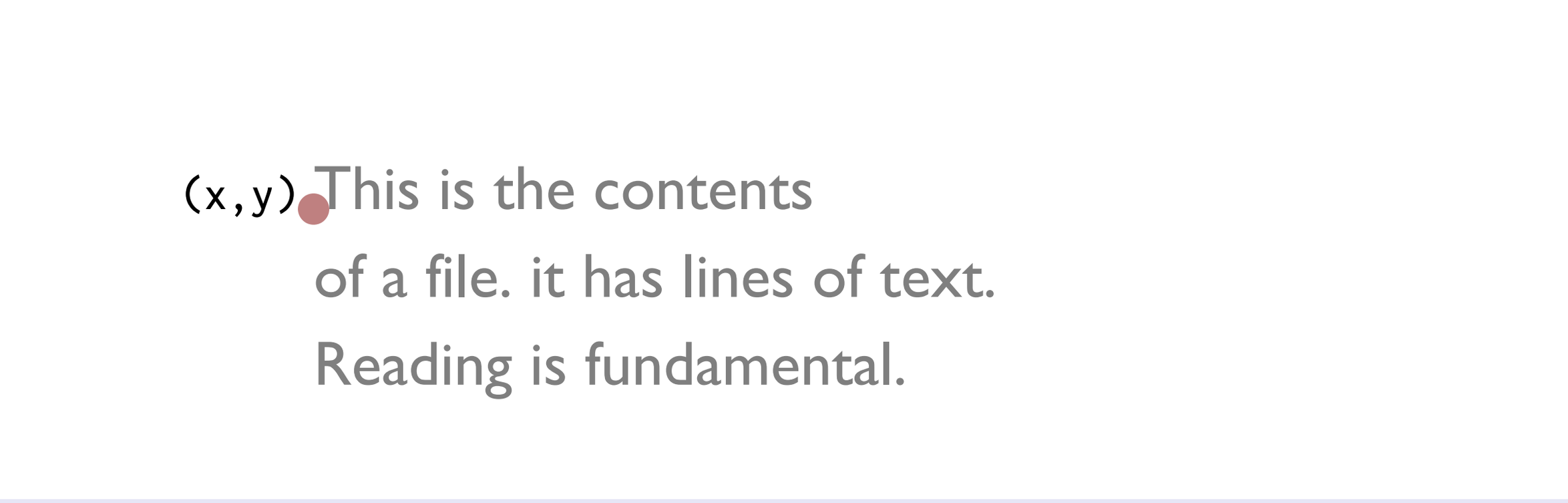

```
textblock "..." x y w fontsize font color op link
```

“Where justice is denied, where poverty is enforced, where ignorance prevails, and where any one class is made to feel that society is an organized conspiracy to oppress, rob and degrade them, neither persons nor property will be safe.”

```
textblock "..." 10 35 30 2
```

“Where justice is denied, where poverty is enforced, where ignorance prevails, and where any one class is made to feel that society is an organized conspiracy to oppress, rob and degrade them, neither persons nor property will be safe.”

```
textblock "..." 50 35 10 1 "sans" "maroon"
```



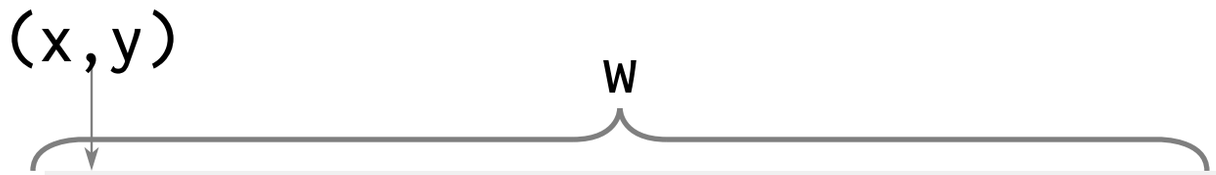

```
textfile "..." x y fontsize font color op link
```

This is the contents
of a file. it has lines of text.
Reading is fundamental.

```
import "fmt"
func main() {
  fmt.Println("Go")
}
```

```
textfile "example.txt" 10 35 2
```

```
textfile "hw-go" 55 35 1.6 "mono" "maroon"
```




```
import "fmt"
func main() {
  fmt.Println("Go")
}
```

## textcode "..." x y w fontsize font color

```
import "fmt"
func main() {
  fmt.Println("Go")
}
```

textcode "hw-go" 10 35 25 1.0

```
import "fmt"
func main() {
  fmt.Println("Go")
}
```

textcode "hw-go" 55 35 40 1.6 "maroon"

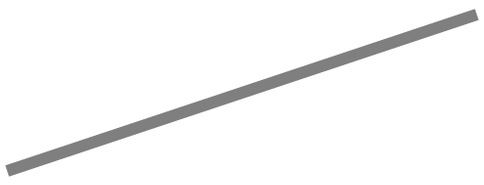
# Graphical Elements

| description | keyword | mandatory | optional |
|---|---|---|---|
| Line | line | x1 y1 x2 y2 lw | color op |
| Horizontal line | hline | x y w | lw color op |
| Vertical line | vline | x y h | lw color op |
| Elliptical arc | arc | x y w h a1 a2 | lw color op |
| Quadratic Bezier | curve | bx by cx cy ex ey | lw color op |
| Circle | circle | x y w | color op |
| Area circle | acircle | x y area | color op |
| Ellipse | ellipse | x y w h | color op |
| Square | square | x y w | color op |
| Rectangle | rect | x y w h | color op |
| Rounded rectangle | rrect | x y w h radius | color |
| Pill shape | pill | x y w h | color |
| Polygon | polygon | "x1 x2...xn" "y1 y2...yn" | color op |
| Polyline | polyline | "x1 x2...xn" "y1 y2...yn" | lw color op |
| N-sided star | star | x y sides inner outer | color op |

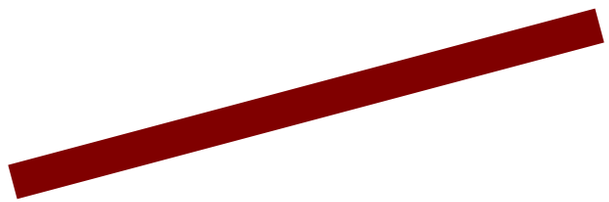

```
line x1 y1 x2 y2 lw  color op
```



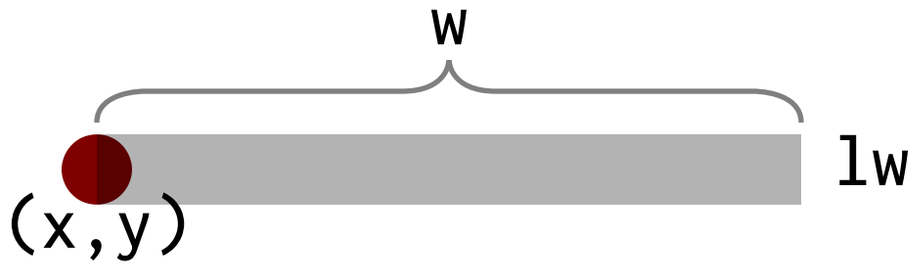
line 10 20 30 20



line 40 20 60 30 0.5



line 70 20 95 30 1.5 "maroon"




```
hline x y w lw color op
```

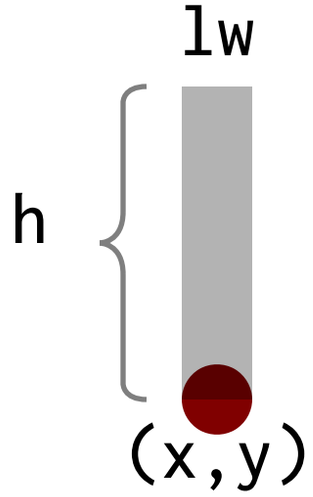


```
hline 15 20 10
```

```
hline 40 20 20 1
```



```
hline 70 20 20 5 "maroon" 20
```




## vline x y w lw color op

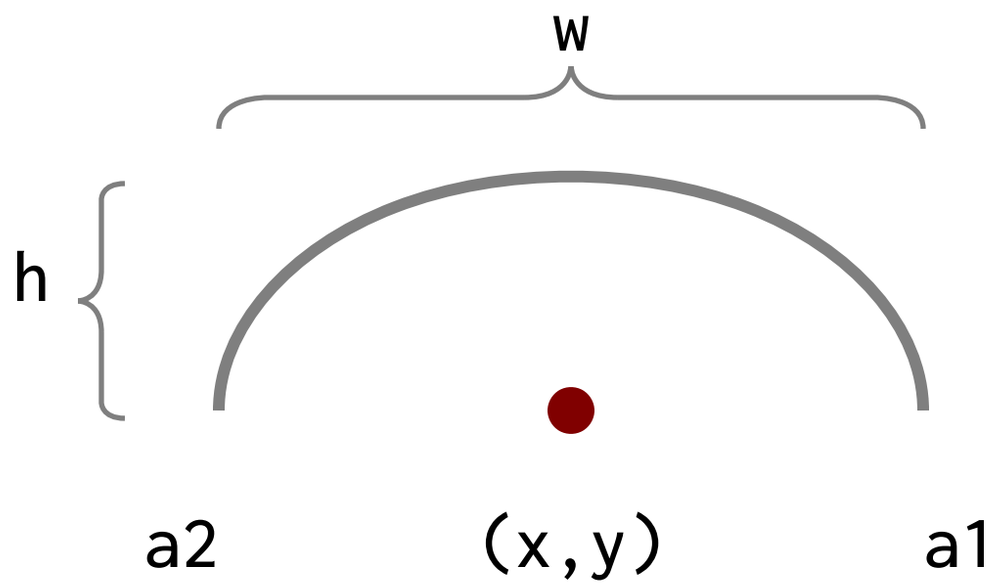


vline 20 20 15

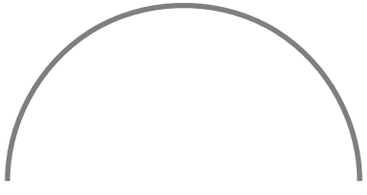
vline 50 20 15 2



vline 80 20 15 10 "maroon" 20

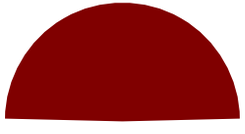



```
arc x y w h a1 a2 lw color op
```

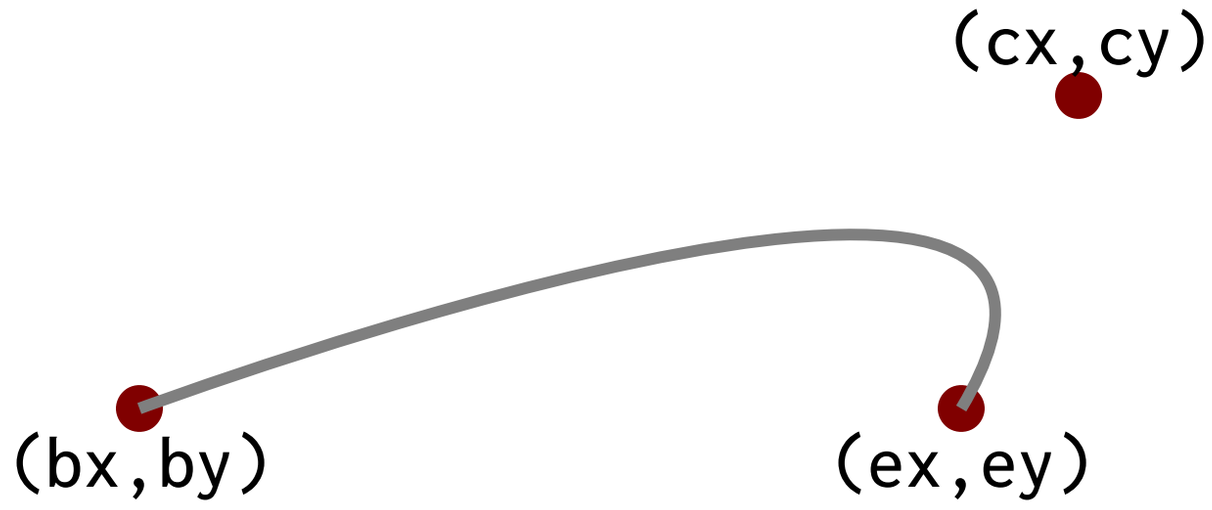


arc 20 20 15 15 0 180

arc 50 20 15 15 0 90 1



arc 80 20 5 5 0 180 5 "maroon"

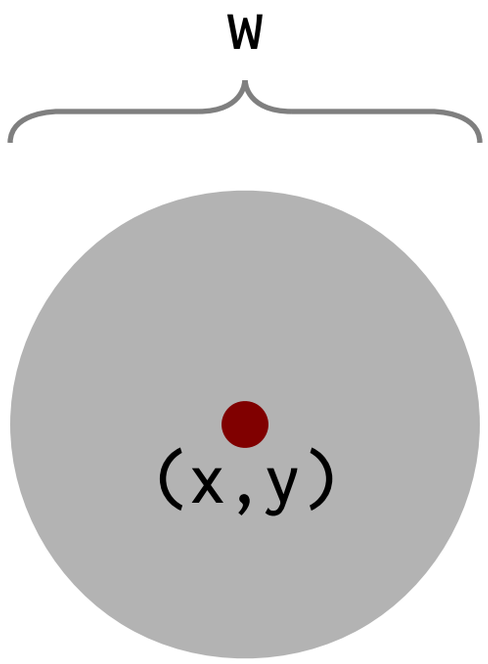



curve bx by cx cy ex ey lw color op

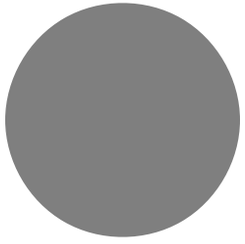
curve 15 20 25 30 30 25

curve 15 20 25 30 30 25

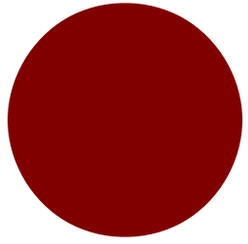
curve 70 20 70 30 90 25 0.5 "maroon"




```
circle x y w  color op
```



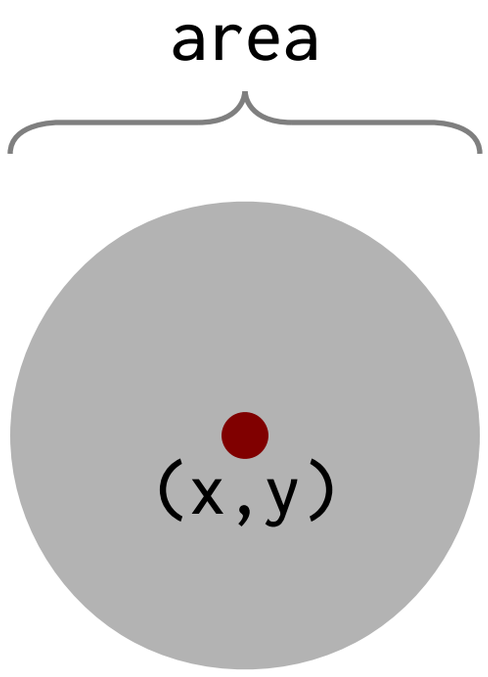
circle 20 20 10



circle 50 20 10 "maroon"



circle 80 20 5 "maroon" 20



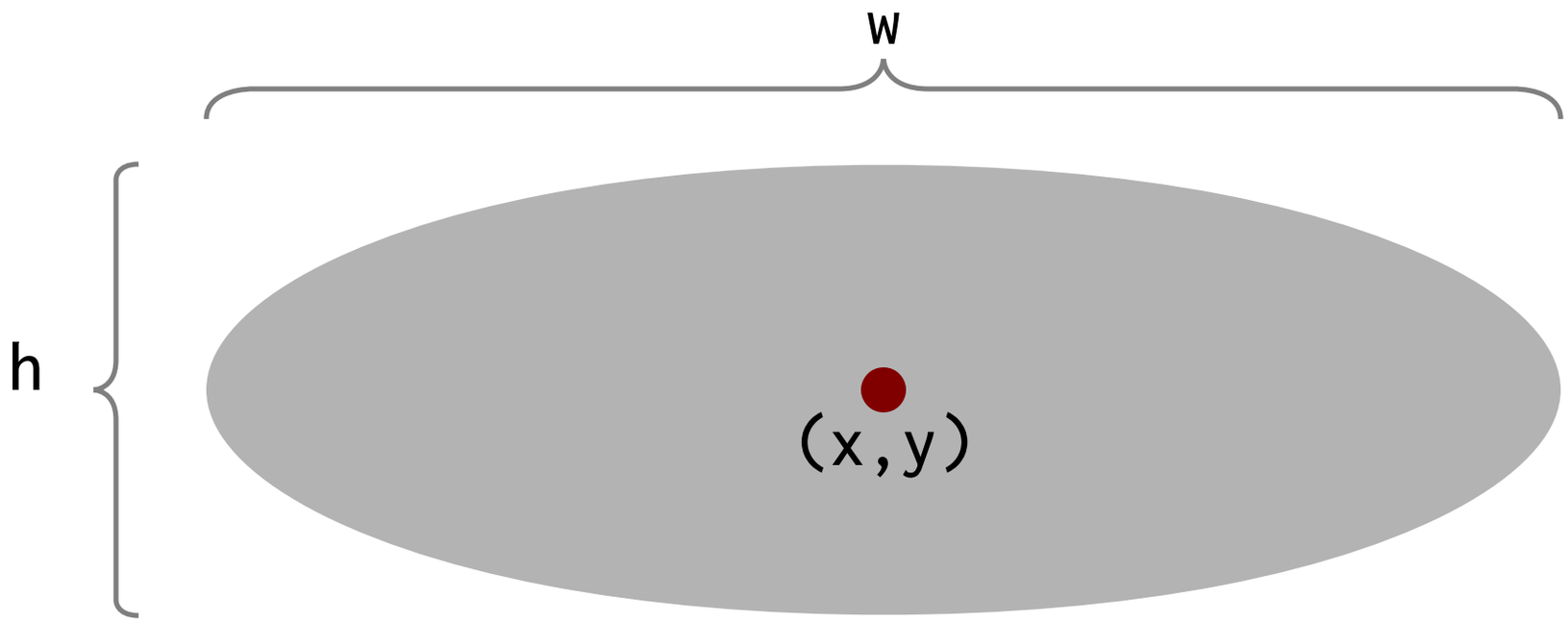

```
circle x y area color op
```

```
acircle 20 20 10
```

```
acircle 50 20 10 "maroon"
```



```
acircle 80 20 5 "maroon" 20
```



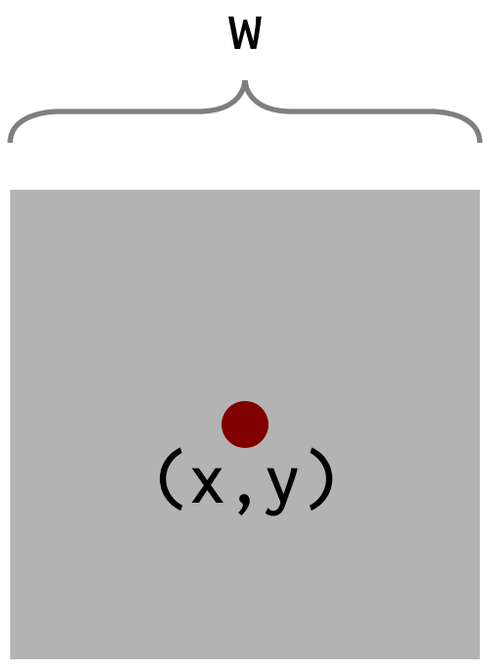

```
ellipse x y w h color op
```

```
ellipse 20 20 10 5
```

```
ellipse 50 20 10 5 "maroon"
```



```
ellipse 80 20 5 10 "maroon" 20
```



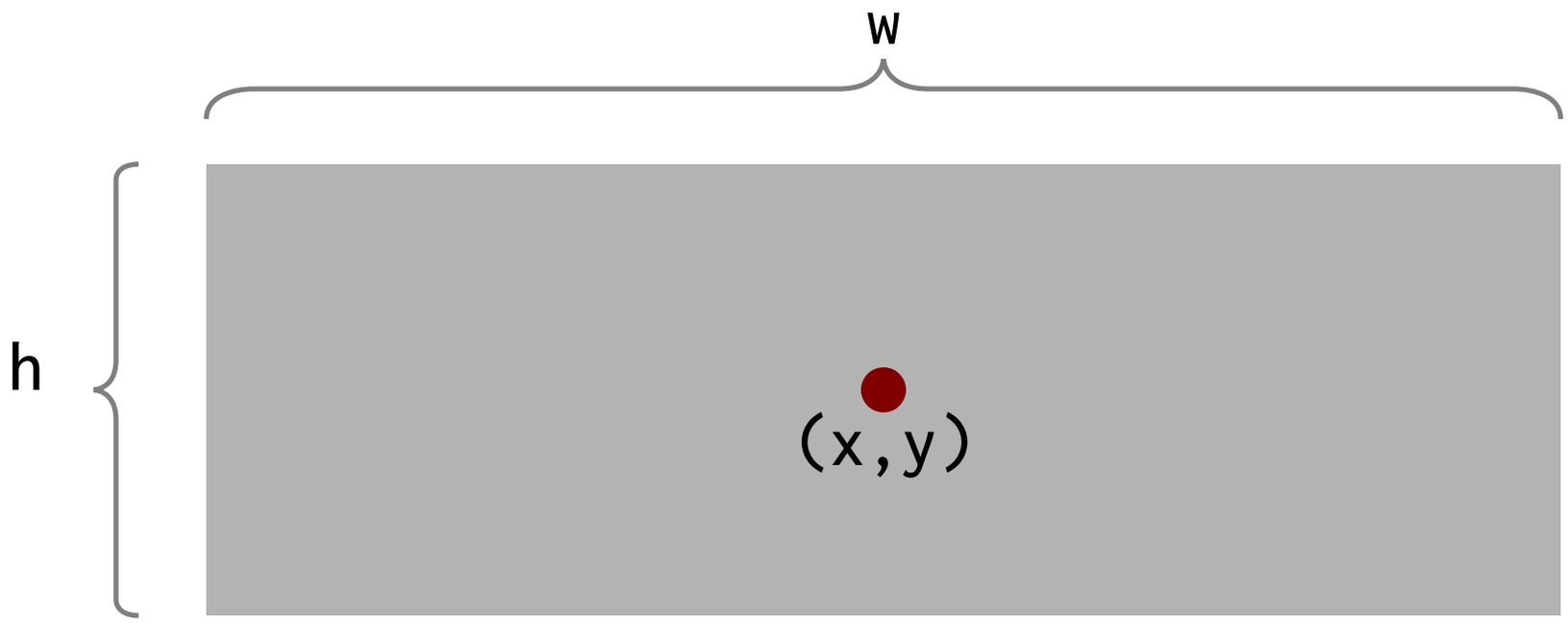

```
square x y w color op
```

square 20 20 10

square 50 20 10 "maroon"



square 80 20 5 "maroon" 20



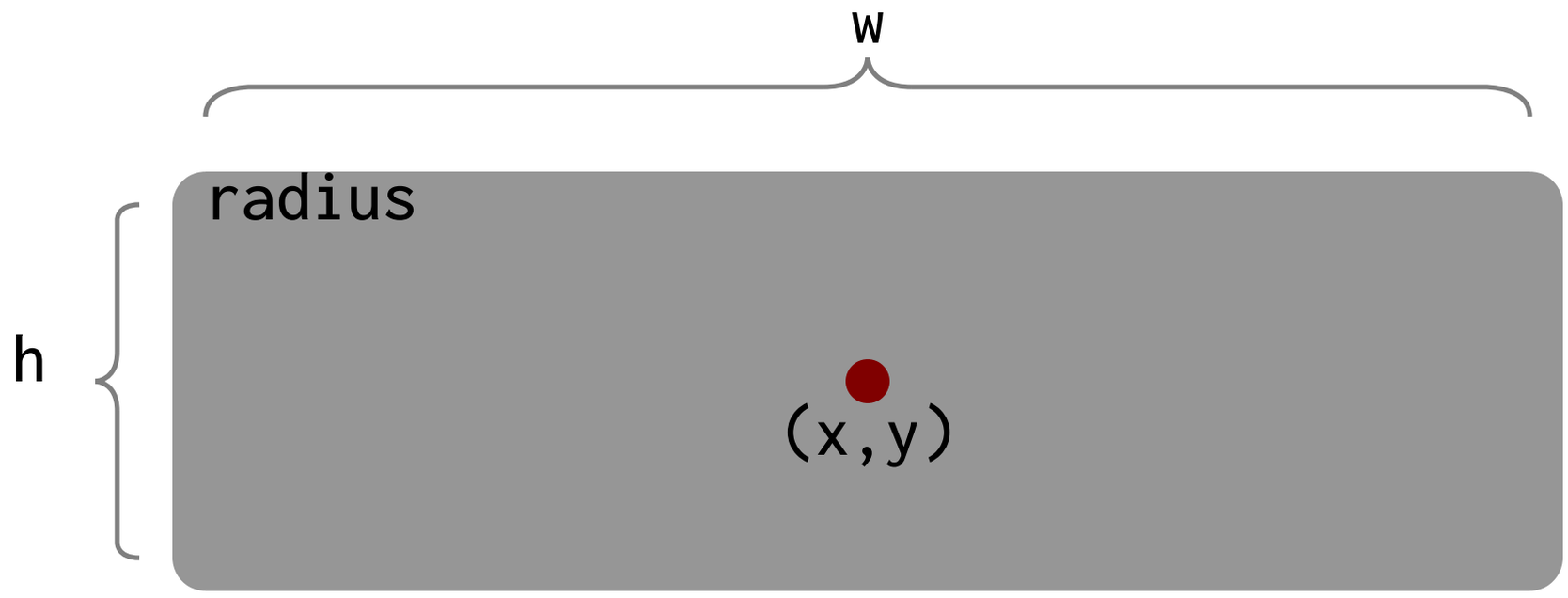

```
rect x y w h color op
```

```
rect 20 20 10 5
```

```
rect 50 20 10 5 "maroon"
```

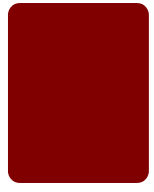


```
rect 80 20 5 10 "maroon" 20
```

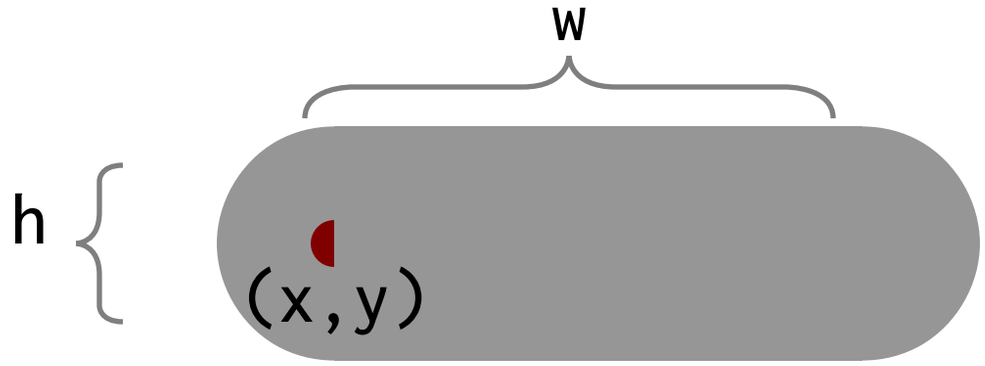



```
rrect x y w h radius color op
```

```
rrect 20 20 10 5 1
```

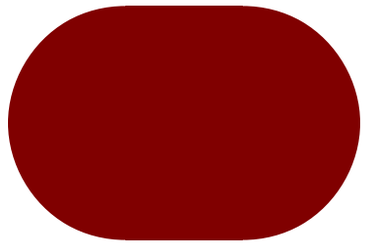


```
rrect 80 20 5 10 1 "maroon"
```

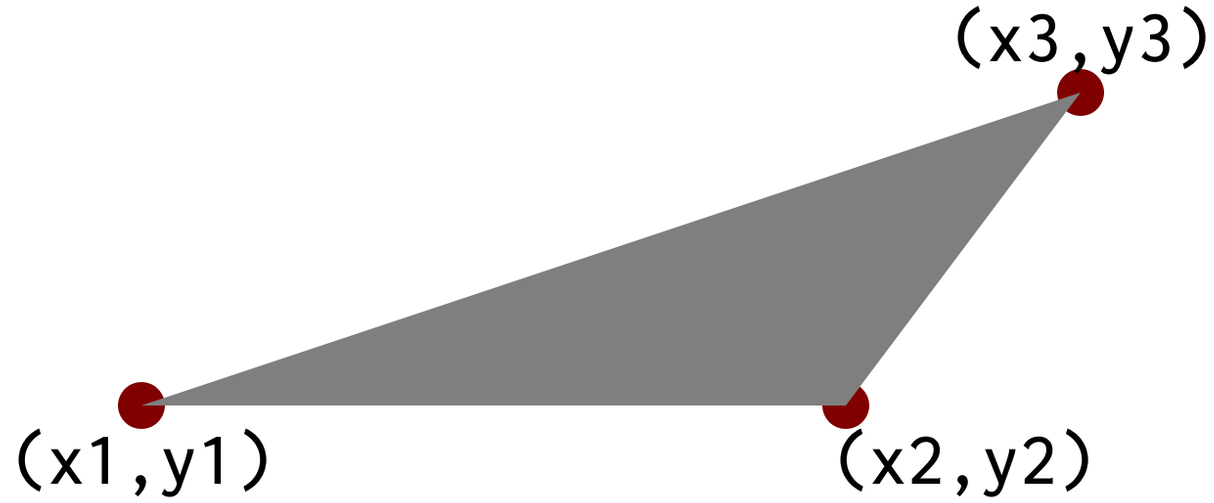



```
pill x y w h color
```

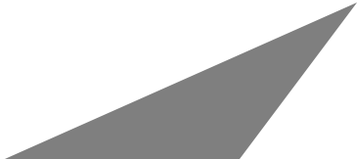
```
pill 20 20 10 5
```



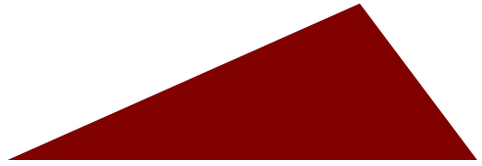
```
pill 80 20 5 10 "maroon"
```



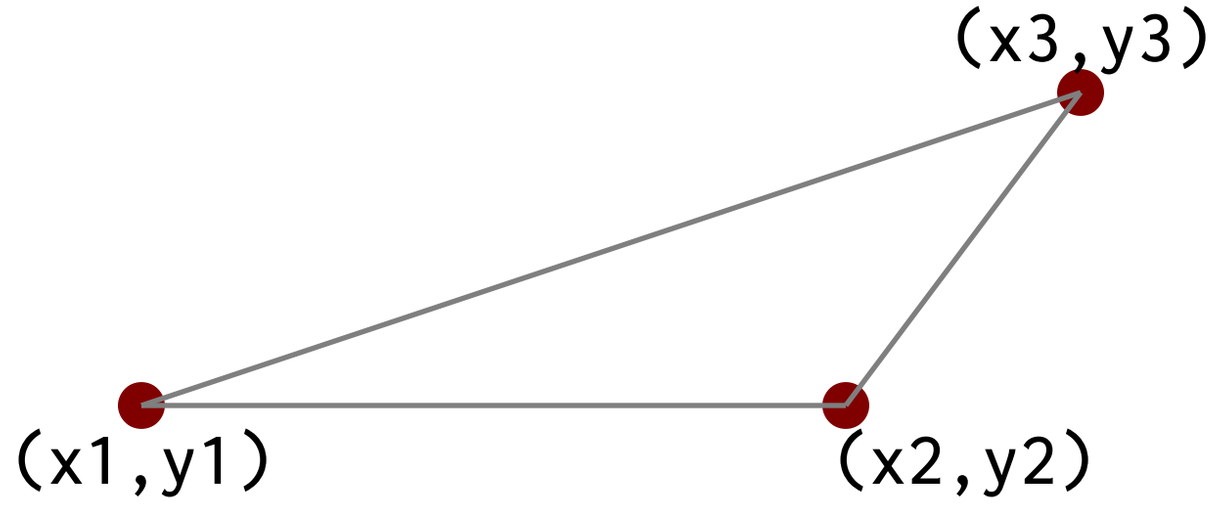

```
polygon "x1 x2...xn" "y1 y2...yn" color op
```



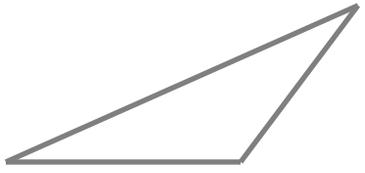
```
polygon "10 25 20" "20 30 20"
```



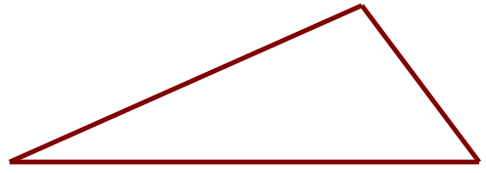
```
polygon "70 85 90" "20 30 20" "maroon"
```



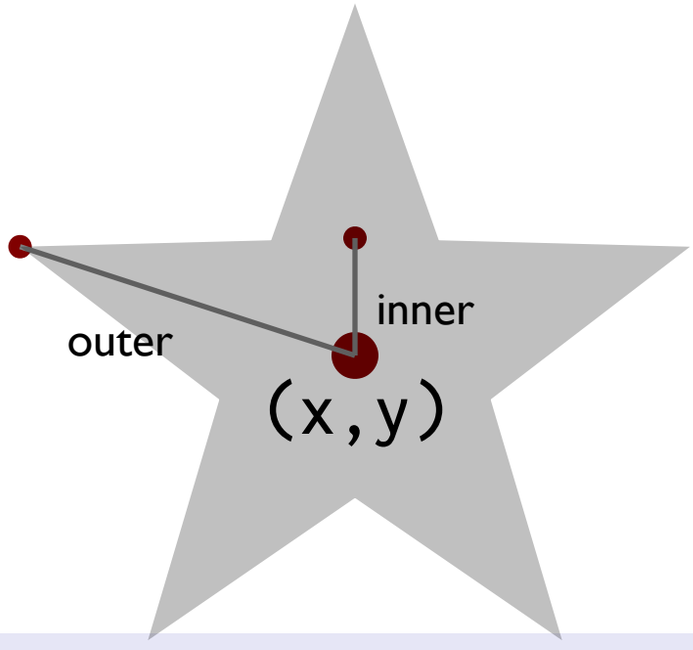

```
polyline "x1 x2...xn" "y1 y2...yn" lw color op
```



```
polyline "10 25 20" "20 30 20"
```

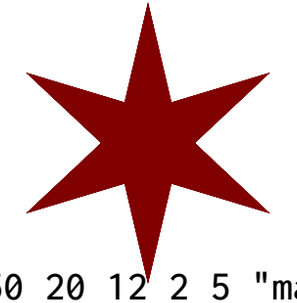


```
polyline "70 85 90" "20 30 20" 0.2 "maroon"
```




```
star x y sides inner outer color op
```

```
star 20 20 5 2 6
```



```
star 50 20 12 2 5 "maroon"
```

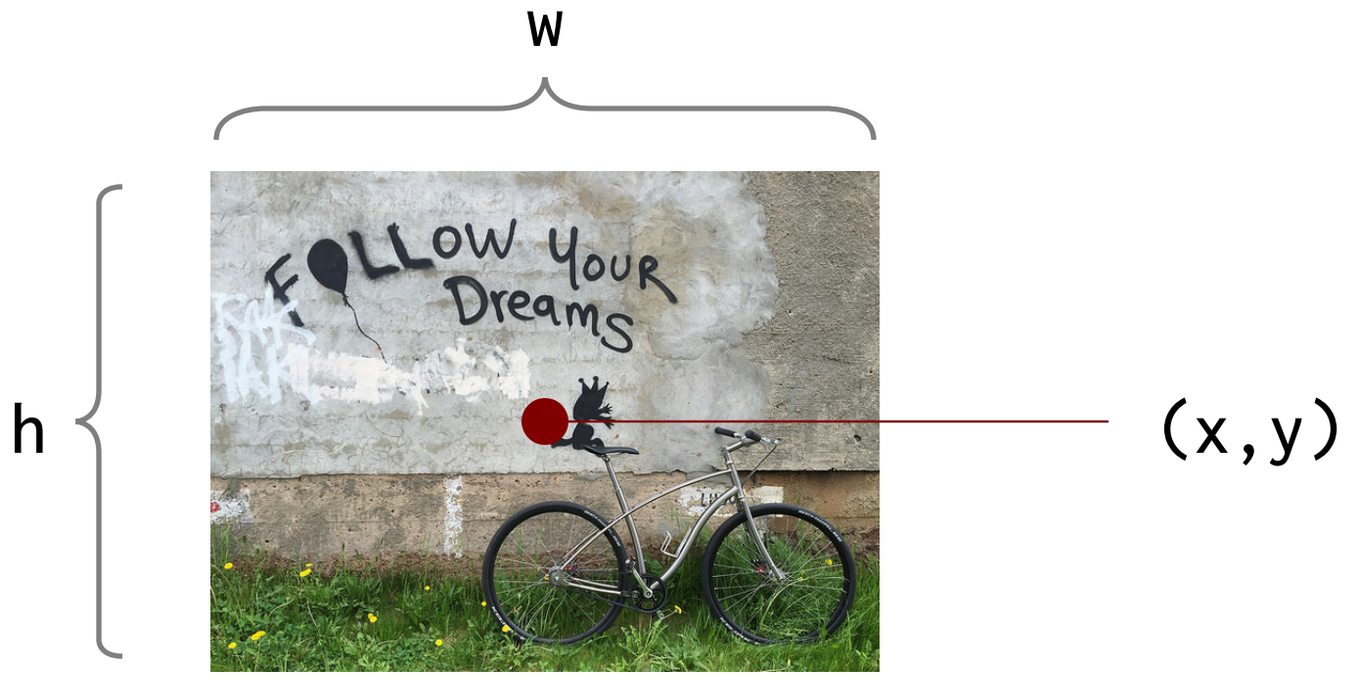
```
star 80 ey 24 2 8 "maroon" 20
```

# Images

| description | keyword | mandatory | optional |
|---|---|---|---|
| Image | `image` | `"file" x y w h` | `scale "link"` |
| Captioned image | `cimage` | `"file" "caption" x y w h` | `scale "link" capsize` |

If h = 0, w specifies the image width in terms of canvas width.

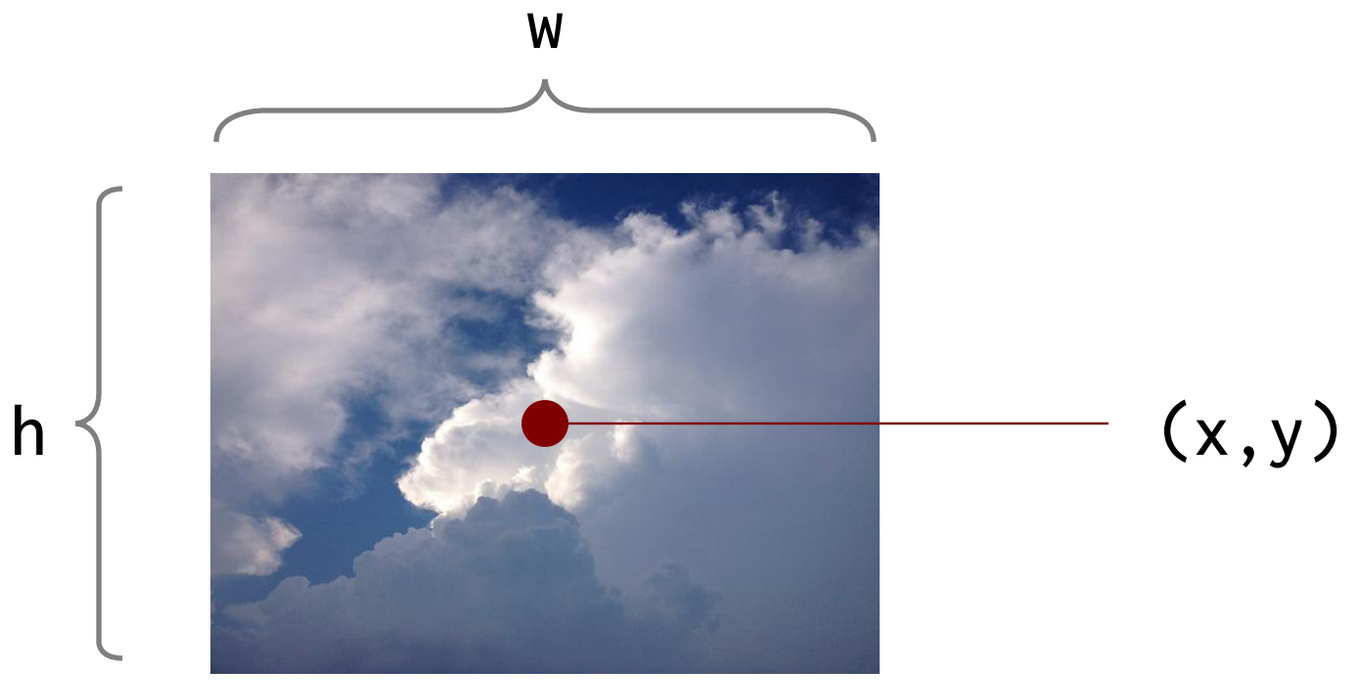
The scale value is a percentage from 1-100, and link is a URL. capsize is the text size of the caption




```
image "file" x y w h scale link
```

image "follow.jpg" 20 25 640 480 10

image "follow.jpg" 75 25 640 480 30




sky

```
cimage "file" x y w h scale link
```

sky

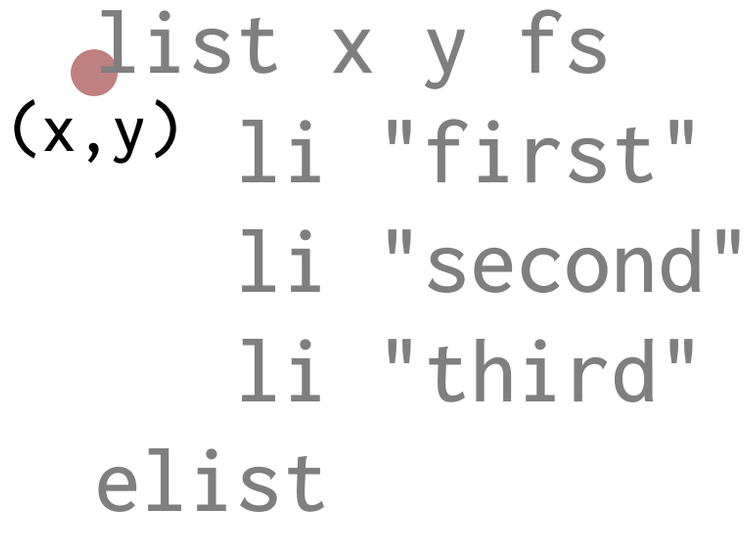
sky

```
cimage "cloudy.jpg" "sky" 20 25 640 480 10
```

```
cimage "cloudy.jpg" "sky" 75 25 640 480 30 "" 1.5
```

# Lists

| description | keyword | mandatory | optional |
|---|---|---|---|
| Plain list | `list` | `x y fontsize` | `font color op spacing` |
| Bullet list | `blist` | `x y fontsize` | `font color op spacing` |
| Numbered list | `nlist` | `x y fontsize` | `font color op spacing` |
| Centered list | `clist` | `x y fontsize` | `font color op spacing` |




## list x y fontsize font color op spacing

```
list 20 30 2.5
    li "one"
    li "two"
    li "three"
elist
```

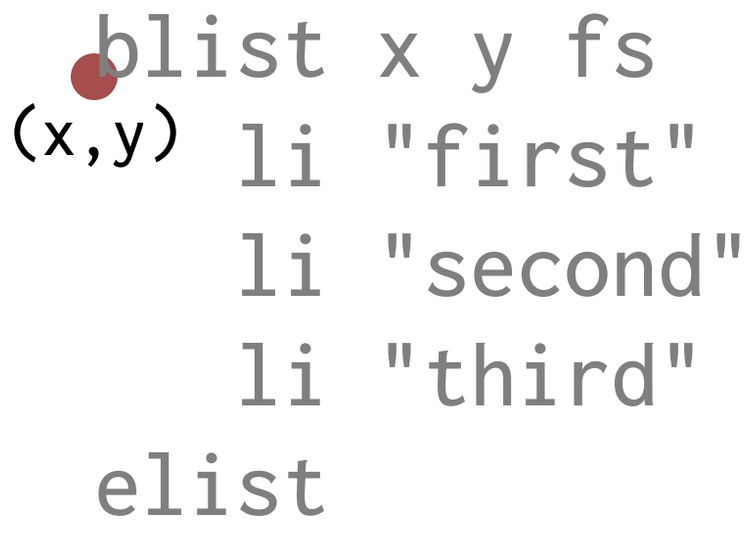
one

two

three

```
list 85 30 2.5 "serif" "maroon" 100 1.0
    li "one"
    li "two"
    li "three"
elist
```

one
two
three



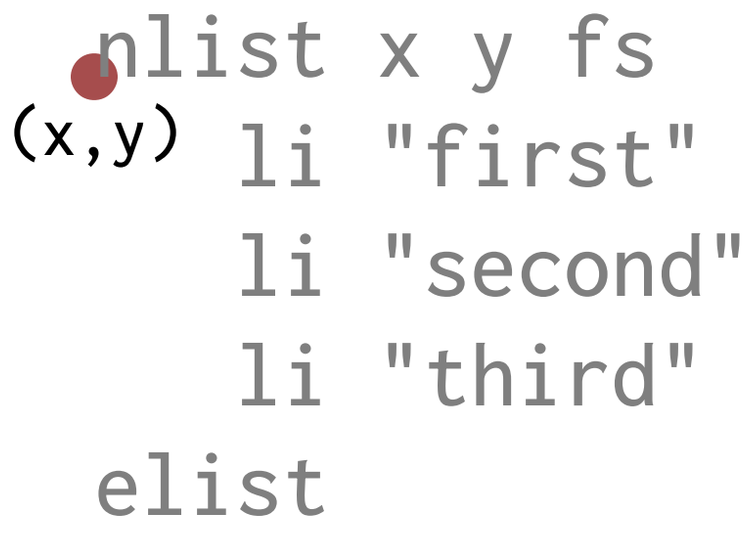

**blist** x y **fontsize** font color op spacing

```
blist 20 30 2.5
    li "one"
    li "two"
    li "three"
elist
```

- one
- two
- three

```
blist 85 30 2.5 "serif" "maroon" 100 1.0
    li "one"
    li "two"
    li "three"
elist
```

- one
- two
- three



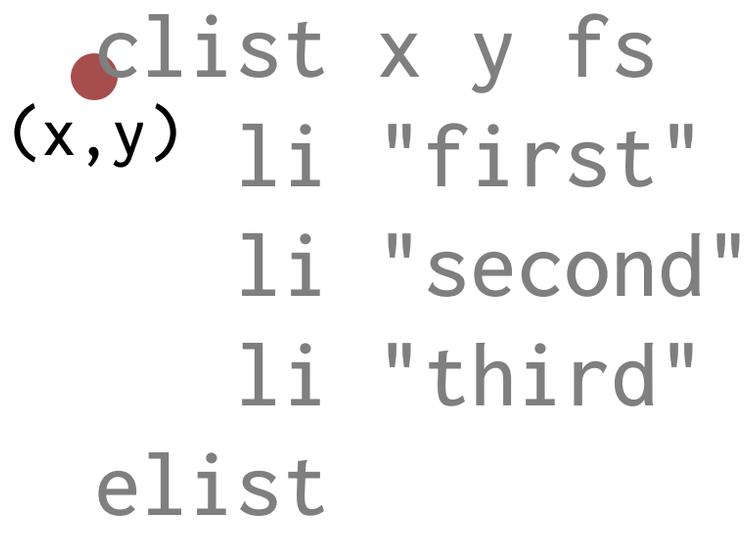

**nlist** x y **fontsize** font color op spacing

```
nlist 20 30 2.5
    li "one"
    li "two"
    li "three"
elist
```

1. one
2. two
3. three

```
nlist 85 30 2.5 "serif" "maroon" 100 1.0
    li "one"
    li "two"
    li "three"
elist
```

1. one
2. two
3. three




clist x y fontsize font color op spacing

```
clist 30 30 2.5
    li "first one"
    li "next"
    li "and last"
elist
```

first one

next

and last

```
clist 90 30 2.5 "serif" "maroon" 100 1.0
    li "first"
    li "next"
    li "and last"
elist
```

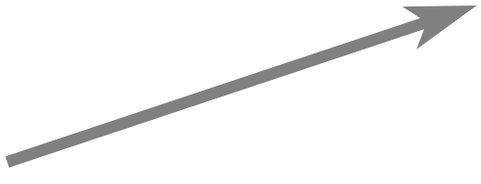
first
next
and last

# Arrows

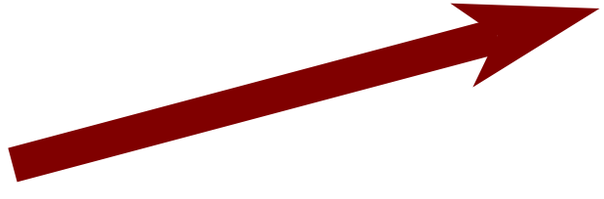
| description | keyword | mandatory | optional |
|---|---|---|---|
| Straight | `arrow` | `x1 y1 x2 y2` | `lw aw ah color op` |
| Left curved | `lcarrow` | `bx by cx cy ex ey` | `lw aw ah color op` |
| Right curved | `rcarrow` | `bx by cx cy ex ey` | `lw aw ah color op` |
| Up curved | `ucarrow` | `bx by cx cy ex ey` | `lw aw ah color op` |
| Down curved | `dcarrow` | `bx by cx cy ex ey` | `lw aw ah color op` |


```
arrow  x1 y1 x2 y2  lw aw ah color op
```

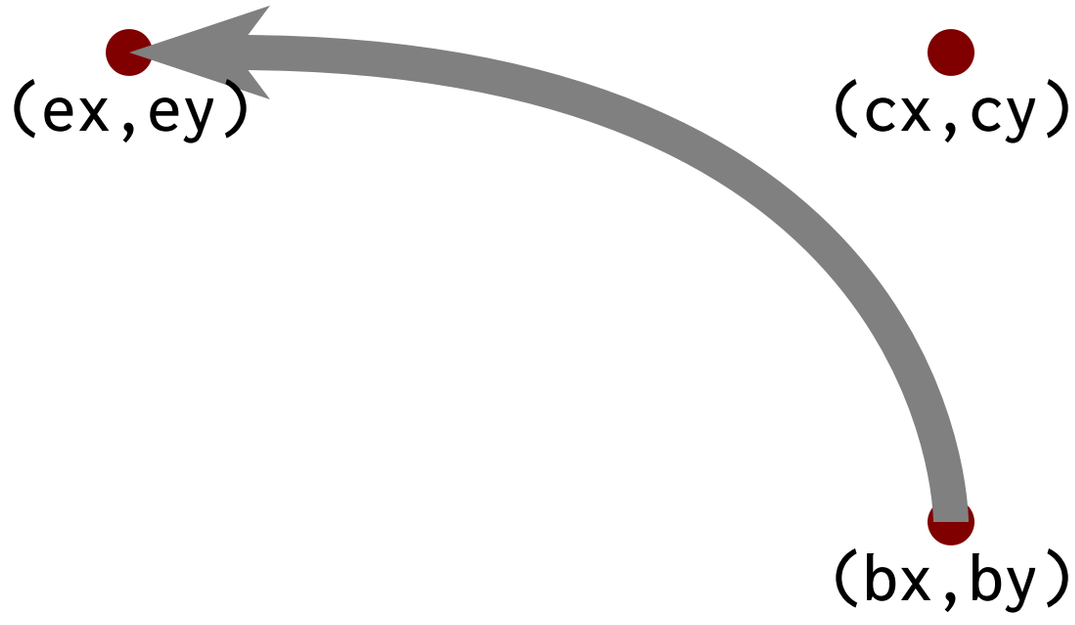
```
arrow 10 20 30 20
```



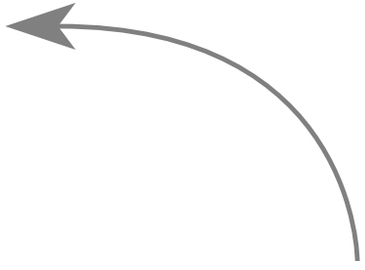
```
arrow 40 20 60 30 0.5
```



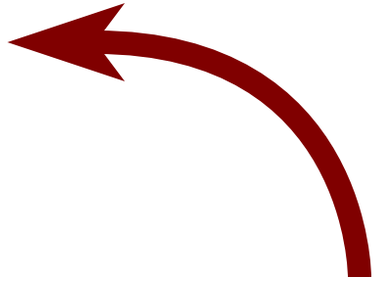
```
arrow 70 20 95 30 1.5 6 6 "maroon"
```



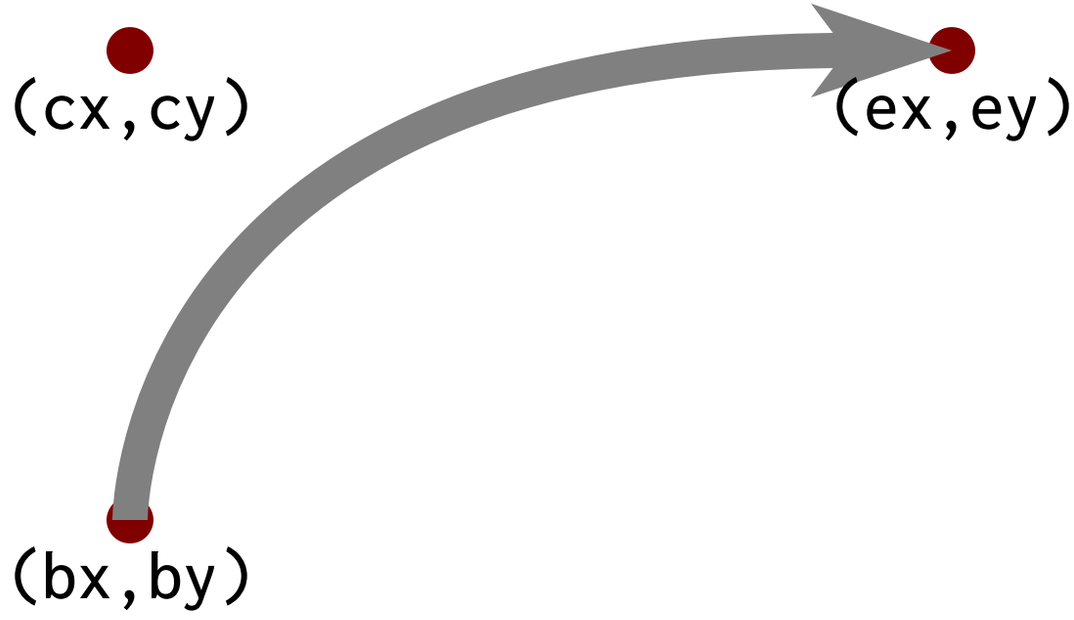

```
lcarrow  bx by cx cy ex ey lw aw ah color op
```



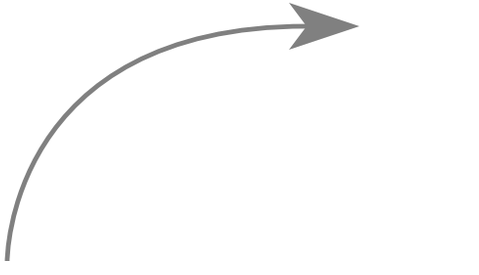
```
lcarrow 30 20 30 35 15 35
```



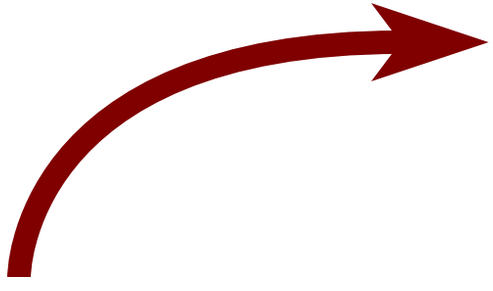
```
lcarrow 70 20 70 35 55 35 1 5 5 "maroon"
```



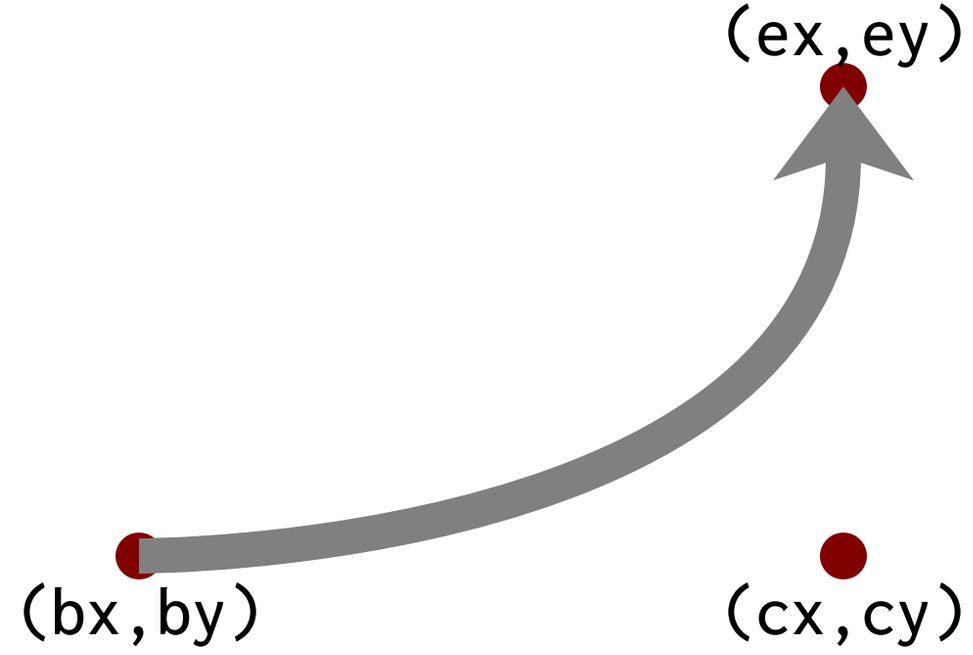

```
rcarrow  bx by cx cy ex ey lw aw ah color op
```



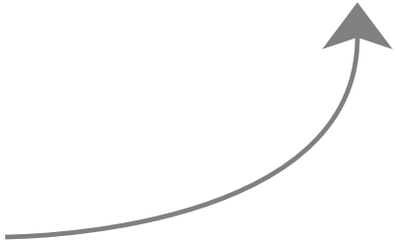
```
rcarrow 15 20 15 35 30 35
```



```
rcarrow 50 20 50 35 70 35 1 5 5 "maroon"
```

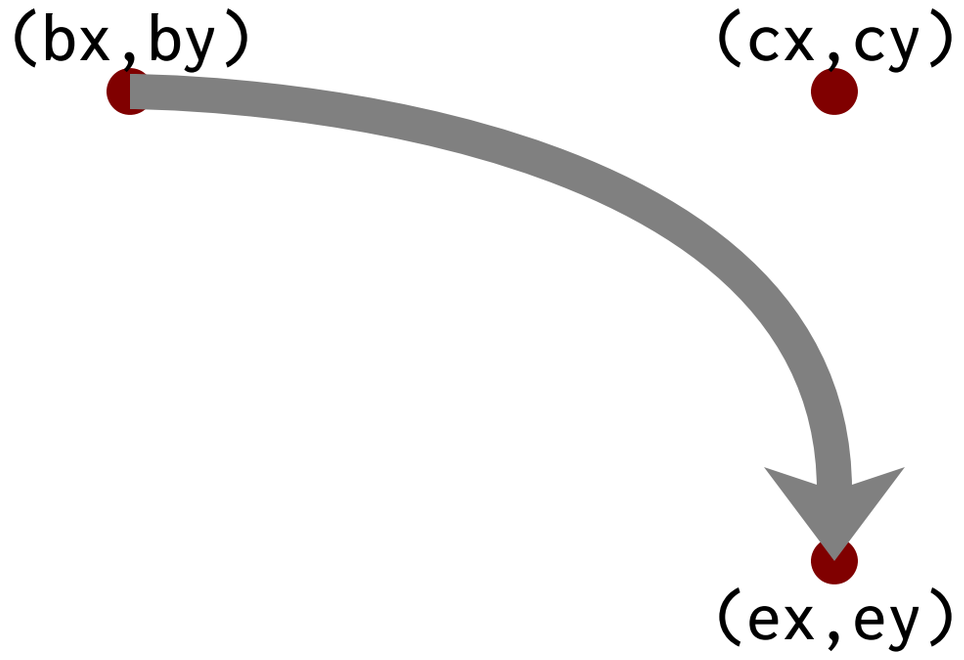



```
ucarrow  bx by cx cy ex ey lw aw ah color op
```



```
ucarrow 15 20 30 20 30 35
```

```
rcarrow 50 20 70 20 70 35 1 5 5 "maroon"
```



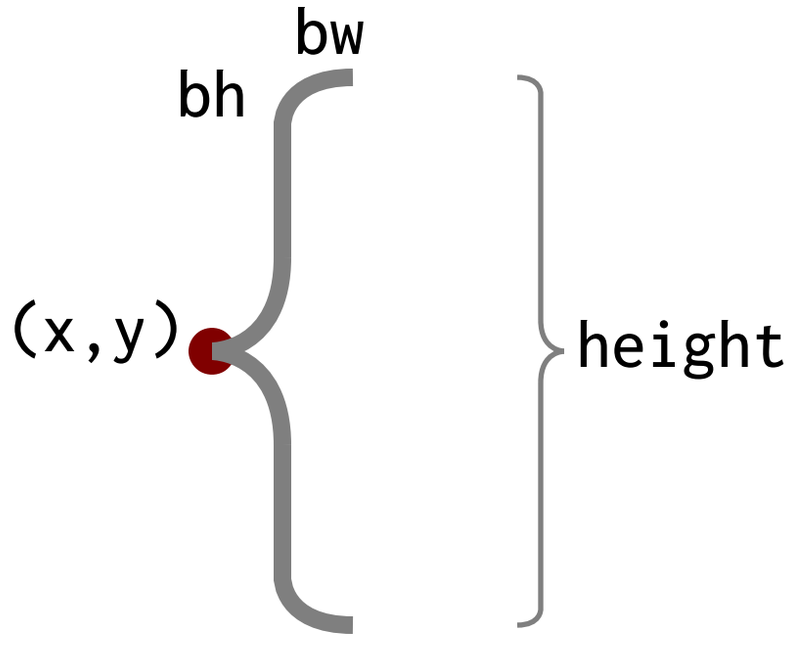

```
dcarrow  bx by cx cy ex ey lw aw ah color op
```

```
dcarrow 15 35 30 30 20
```

```
dcarrow 50 35 70 35 70 20 1 5 5 "maroon"
```

# Braces and Brackets

| description | keyword | mandatory | optional |
|---|---|---|---|
| Left brace | `lbrace` | `x y height bw bh` | `lw color op` |
| Right brace | `rbrace` | `x y height bw bh` | `lw color op` |
| Up brace | `ubrace` | `x y width bw bh` | `lw color op` |
| Down brace | `dbrace` | `x y width bw bh` | `lw color op` |
| Left bracket | `lbracket` | `x y width height` | `lw color op` |
| Right bracket | `rbracket` | `x y width height` | `lw color op` |
| Up bracket | `ubracket` | `x y width height` | `lw color op` |
| Down bracket | `dbracket` | `x y width height` | `lw color op` |

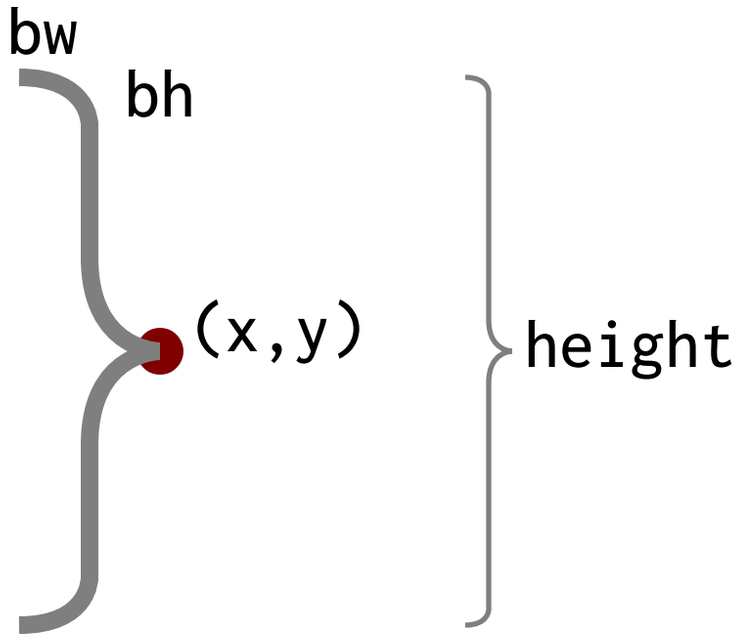



```
lbrace  x y height bw bh   lw color op
```

```
lbrace 20 25 20 2 2
```

```
lbrace 50 25 20 4 4 1
```

```
lbrace 80 25 20 6 3 1 "maroon"
```

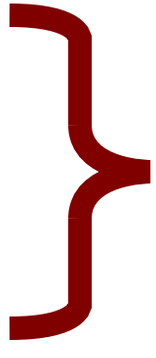



```
rbrace  x y height bw bh   lw color op
```

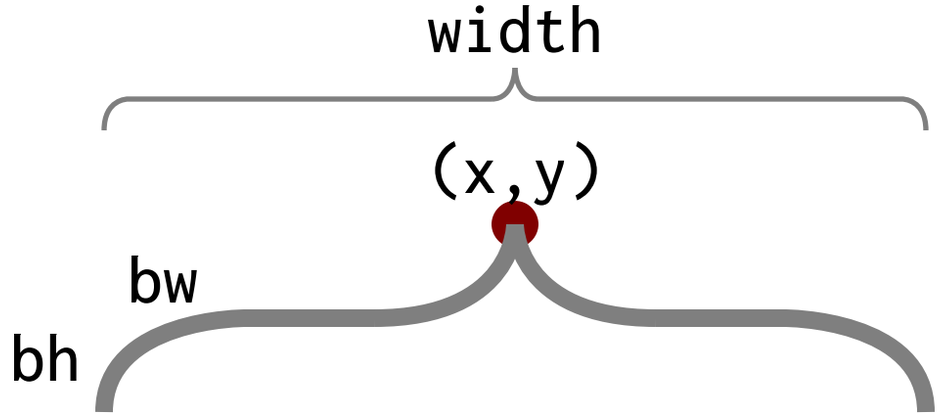
```
rbrace 20 25 20 2 2
```



```
rbrace 50 25 20 4 4 1
```



```
rbrace 80 25 20 6 3 1 "maroon"
```

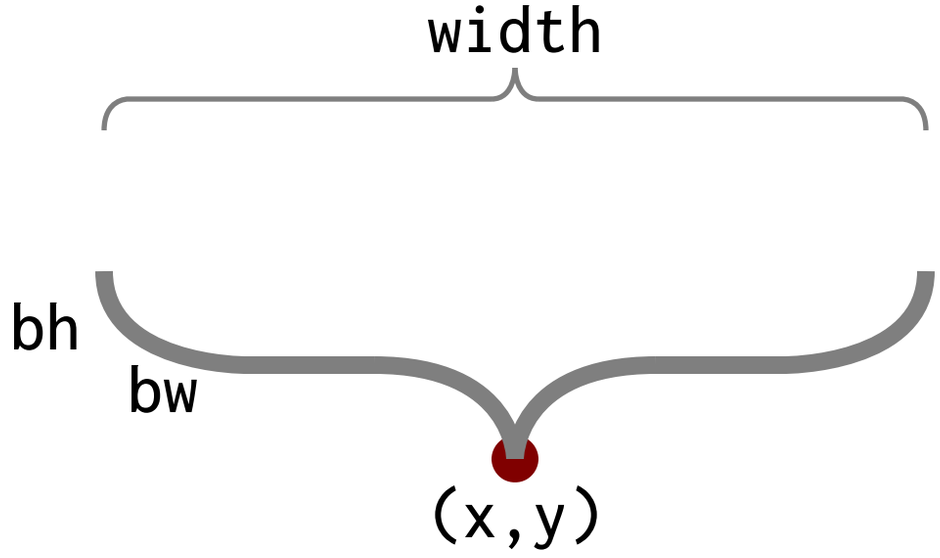



ubrace x y width bw bh lw color op

ubrace 20 25 20 2 4

ubrace 50 25 20 4 8 1

ubrace 80 25 20 4 8 1 "maroon"

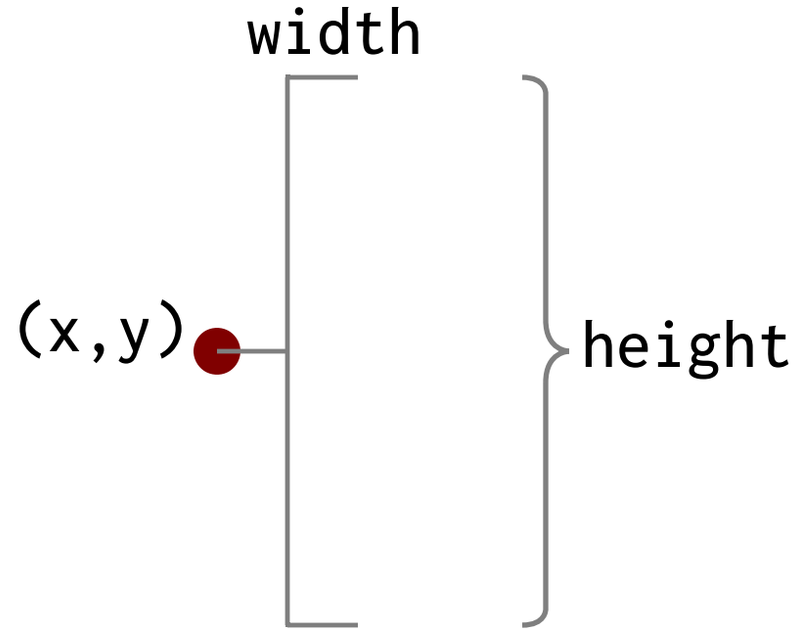



dbrace x y width bw bh lw color op

dbrace 20 25 20 2 4

dbrace 50 25 20 4 8 1

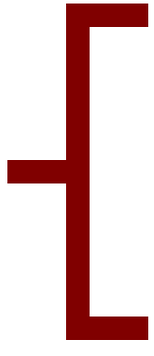
dbrace 80 25 20 4 8 1 "maroon"



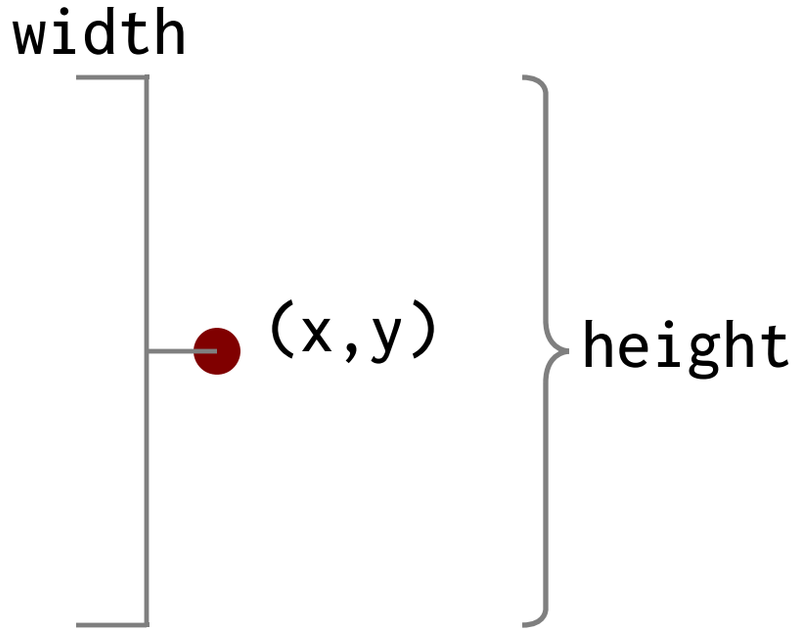

```
lbracket  x y width height   lw color op
```

lbrace 20 25 2 20

lbracket 50 25 4 20 1



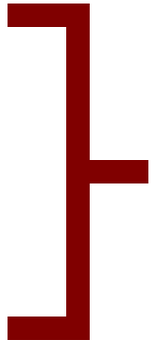
lbracket 80 25 6 20 1 "maroon"



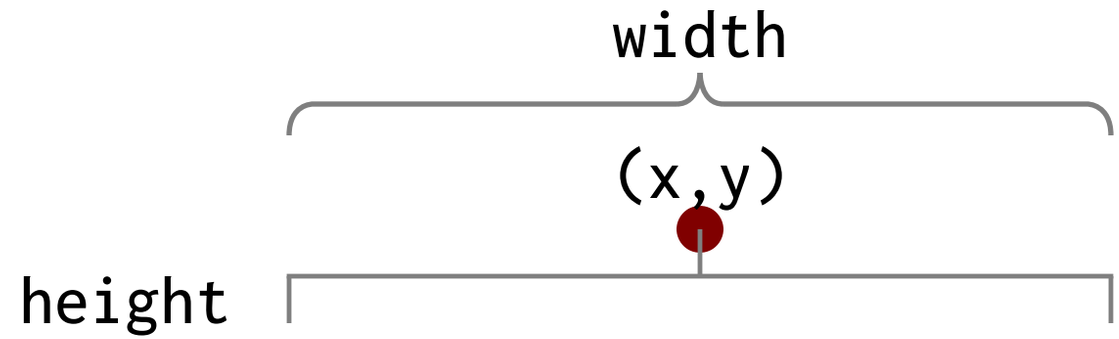

```
rbracket  x y width height   lw color op
```

lbrace 20 25 2 20

rbracket 50 25 4 20 1



rbracket 80 25 6 20 1 "maroon"

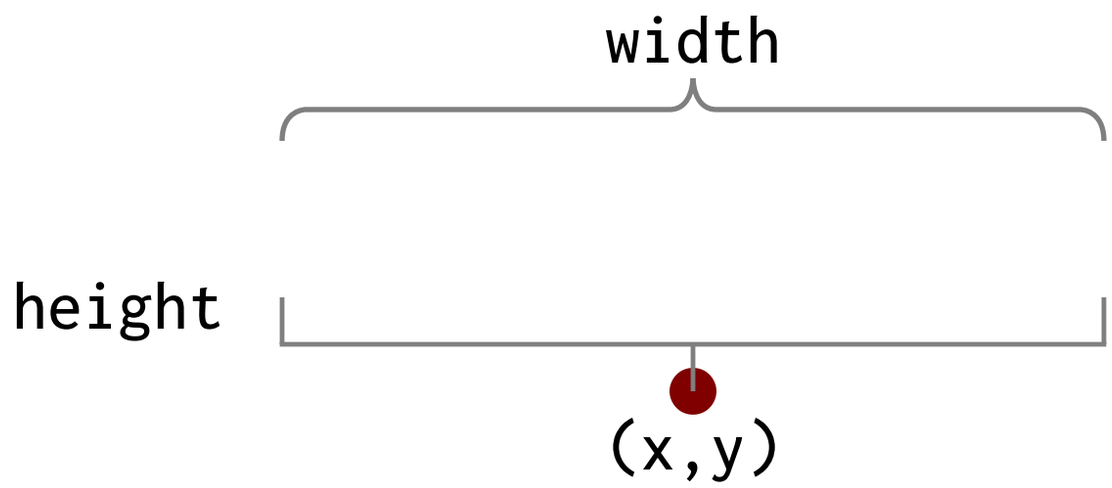



```
ubracket  x y width height   lw color op
```

ubracket 20 25 20 4

ubracket 50 25 20 4 0.3

ubracket 80 25 20 4 1 "maroon"




```
dbracket  x y width height   lw color op
```

dbracket 20 25 20 4

dbracket 50 25 20 4 0.3

dbracket 80 25 20 4 1 "maroon"

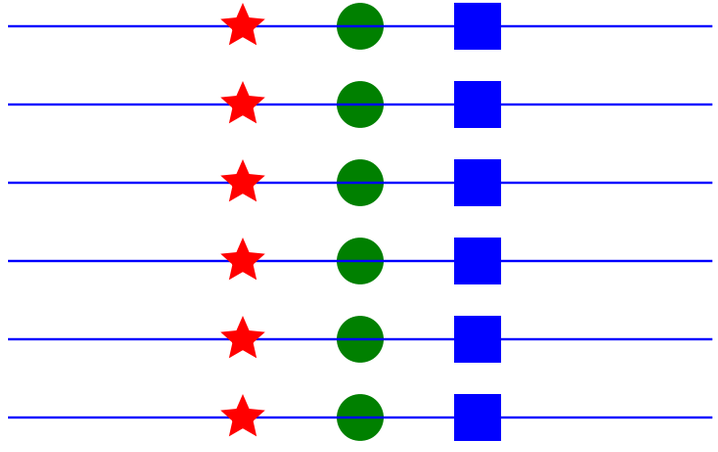
# Loop, If, Built-ins

| description | keyword | mandatory |
|---|---|---|
| Loop | `for v=` | `begin end [increment] ... efor` |
| Conditional | `if` | `condition ... [else] ... eif` |
| Polar coordinate (x) | `x=polarx` | `x y radius angle` |
| Polar coordinate (y) | `y=polary` | `x y radius angle` |
| Polar coordinates | `value=polar` | `x y radius angle` |
| Area | `value=area` | `expression` |
| Formatted text | `value=format` | `fmt expression or up to 5 args` |
| Substring | `value=substr` | `string begin end` |
| Random number | `value=random` | `min max` |
| Value mapping | `value=vmap` | `data min1 max1 min2 max2` |
| Define function | `def` | `name arg1 ... argn ... edef` |
| Import function | `import` | `"file"` |
| In-line data | `data` | `"file" ... edata` |
| Objects on a grid | `grid` | `"file" x y hspace vspace edge` |
| Rulers | `ruler` | `increment [color]` |

```
for v=begin end [increment]
...items to repeat using v
efor
```

## for v=begin end increment ...efor

```
for v=10 35 5
    hline  50 v 30 0.1  "blue"
    star   60 v 5 1 0.4 "red"
    circle 65 v 2 "green"
    square 70 v 2 "blue"
efor
```



```
if condition
...statements when true
else
...statements when false
eif
```

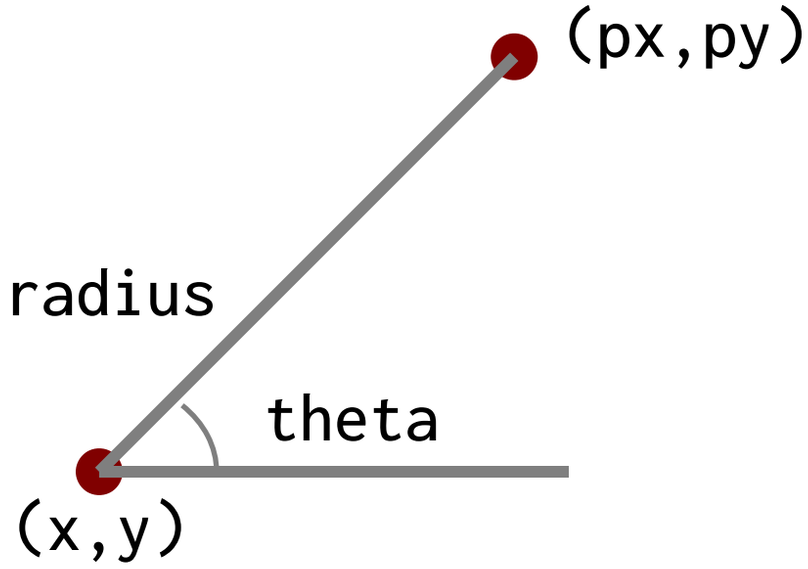
else block may be omitted

## if condition ... else ... eif

| Condition | Description |
|---|---|
| `if a == b` | if a equals b |
| `if a != b` | if a not equal to b |
| `if a > b` | if a less than b |
| `if a < b` | if a greater than b |
| `if a >= b` | if a greater than or equal to b |
| `if a <= b` | if a less than or equal to b |
| `if a >< b c` | if a is between b and c |

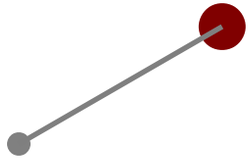
```
a=60
b=3
if a > b
    text "hello" a b 2.5 "sans" "red"
else
    text "bye"   a b 2.5 "sans" "blue"
eif
```

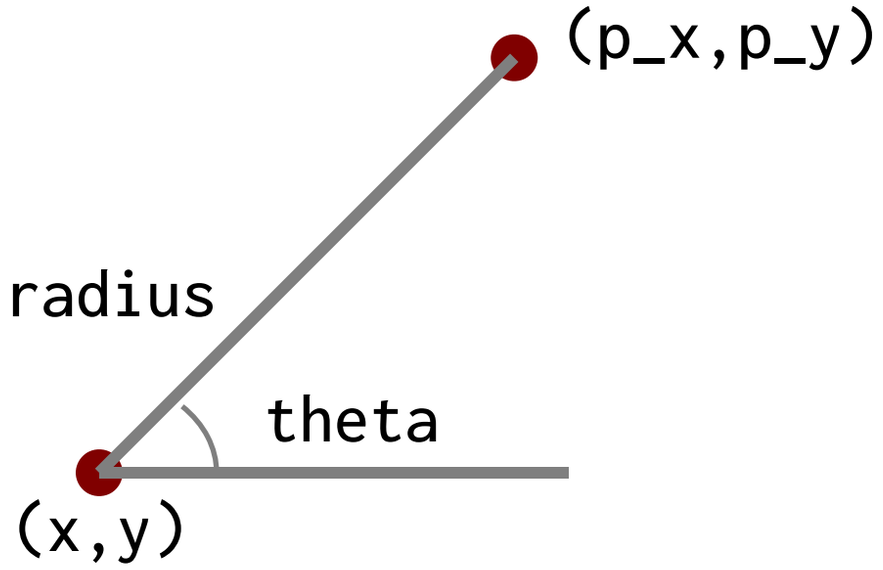
hello




```
px=polarx x y radius theta
py=polary x y radius theta
```

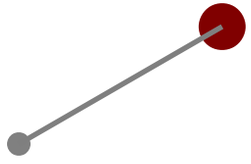
```
cpx=60
cpy=20
px1=polarx cpx cpy 10 30
py1=polary cpx cpy 10 30
line cpx cpy px1 py1
circle cpx cpy 1  "gray"
circle px1 py1 2 "maroon"
```






## p=polar x y radius theta

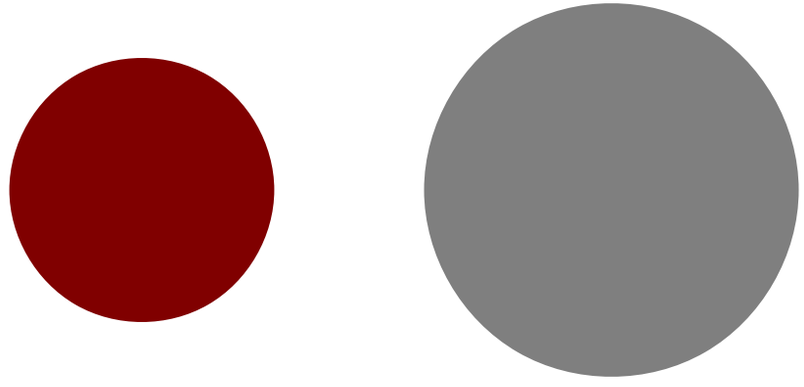
```
cpx=60
cpy=20
point=polar cpx cpy 10 30
line cpx cpy point_x point_y
circle cpx cpy 1  "gray"
circle point_x point_y 2 "maroon"
```



```
v=123.45
a=area v
↑      ↑
area   original value
```

value=area expression

```
m1=100
m2=200
a1=area m1
a2=area m2
circle 60 20 a1 "maroon"
circle 80 20 a2
```



```
x=3.14159
y=2.0
title=format "Value=%.2f" x*y
```

Value=6.28 — format string — expression

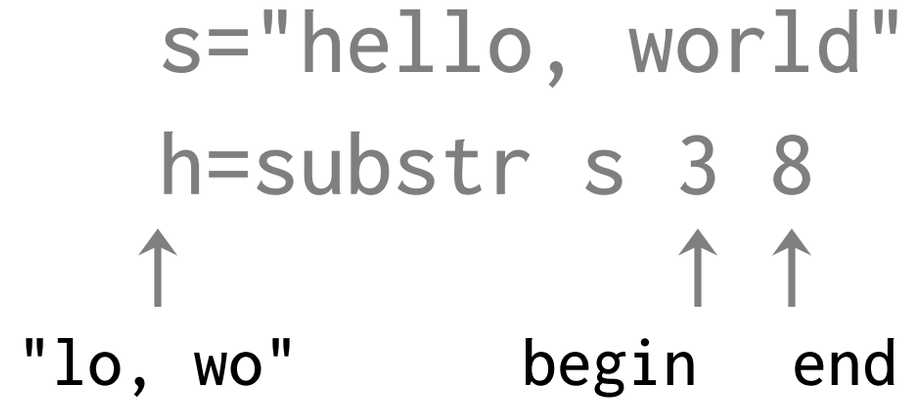
## value=format fmt expression or up to 5 args

```
v1=100.3
v2=200.234
pi=3.1415926
title=format "%.2f Million (USD)" v1
subtitle=format "Total value: %.2f" v1+v2
args=format "Multiple args (%v,%v,%.3f)" 80 10 pi
ctext title    80 30 4 "sans" "maroon"
ctext subtitle 80 20 3 "sans" "gray"
ctext args     80 10 2 "mono"
```

100.30 Million (USD)

Total value: 300.53

Multiple args (80,10,3.142)




value=substr string begin end

```
now="Now is the time for all good men"
s1=substr now 0 14
s2=substr now 16 18
s3=substr now 24 end
ctext s1 70 34 3
ctext s2 70 24 3
ctext s3 70 15 3
```

Now is the time

for

good men

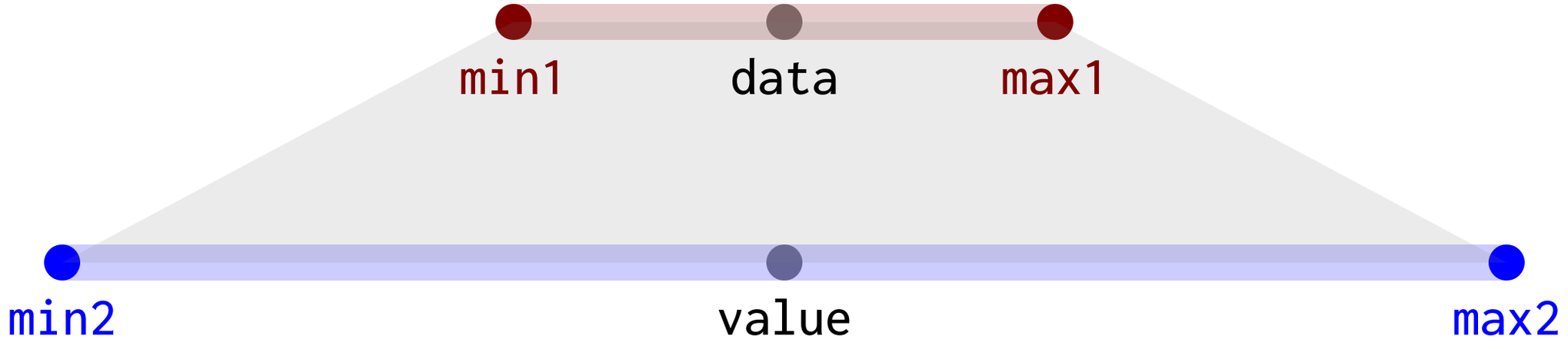

value=random min max

```
rx1=random 5 30
ry1=random 15 35
circle rx1 ry1 3 "maroon"
```



```
rx2=random 40 60
ry2=random 15 35
circle rx2 ry2 3 "green"
```

```
rx1=random 75 95
ry1=random 15 35
circle rx3 ry3 3 "blue"
```




## value=vmap data min1 max1 min2 max2

```
yrmin=1776
yrmax=2021
smin=60
smax=90
vp=vmap 1945 yrmin yrmax smin smax
line   smin 20 smax 20 0.5 "gray" 20
circle smin 20 1
circle smax 20 1
circle vp 20 2 "maroon"
```

# import "doit.dsh"

contents of "doit.dsh"

```
def doit fx fy fs ft
    ctext ft fx fy fs "serif" "purple"
edef
```

doit 50 20 2.5 "hello"

```
fx=50
fy=20
fs=2.5
ft="hello"
ctext ft fx fy fs "serif" "purple"
```

```
import "file"
```

# calling the function

## call again

```
doit 50 30 5 "calling the function"
doit 50 20 4 "call again"
```

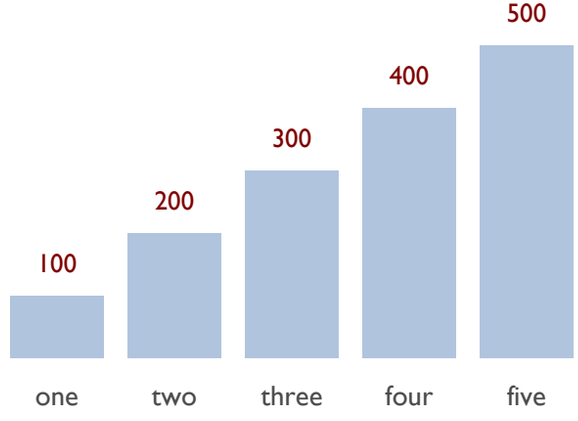
```
data "file.d"
first 20
second 100
third 200
edata
```

data file

data values

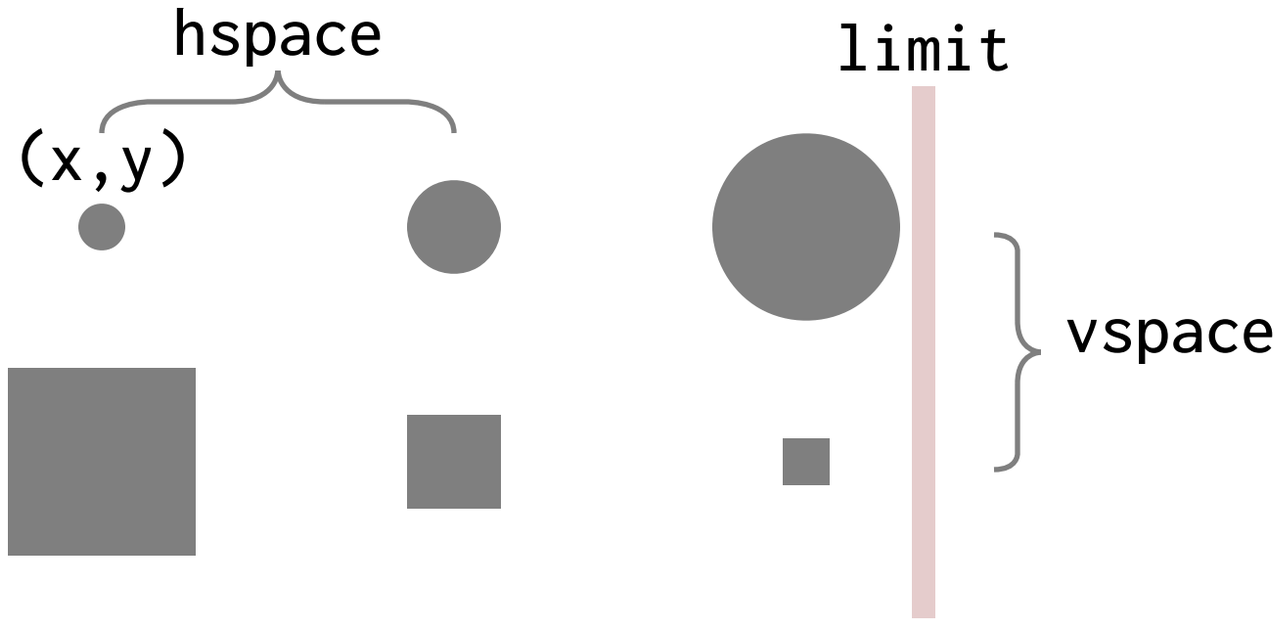
## data "file" ... edata

```
data "test.d"
    one   100
    two   200
    three 300
    four  400
    five  500
edata
dchart -bar -left 50 -bottom 15 -right 70 -top 35 "test.d"
```



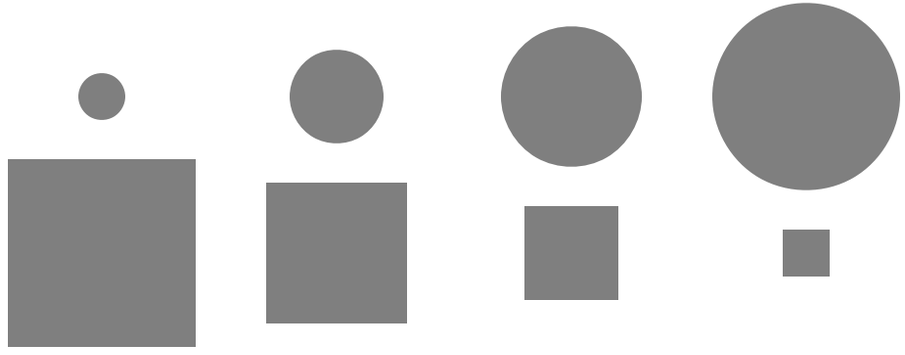
file

```
circle x y 2
circle x y 4
circle x y 8
square x y 8
square x y 4
square x y 2
```

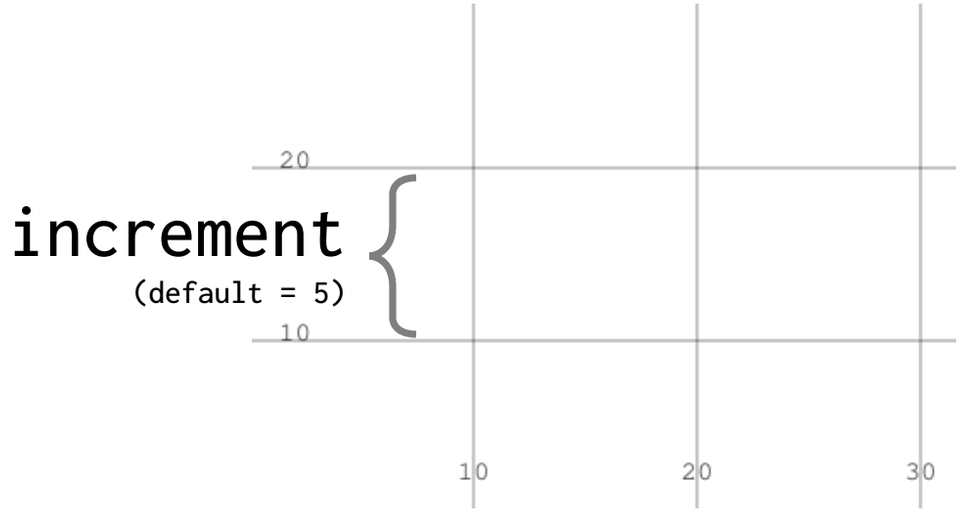



## grid "file" x y hspace vspace limit

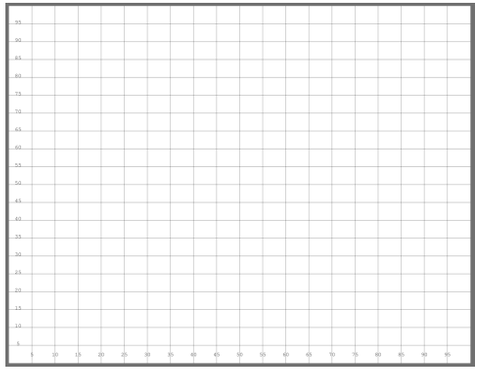
```
circle x y 2
circle x y 4
circle x y 6
circle x y 8
square x y 8
square x y 6
square x y 4
square x y 2
```



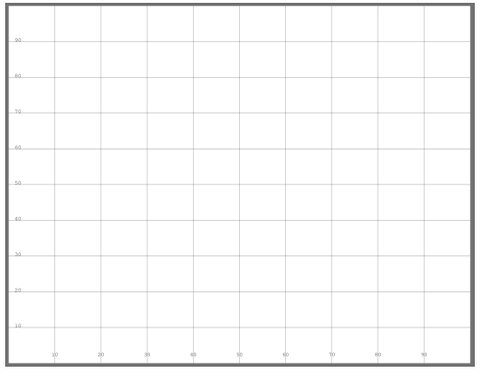
```
grid "code/grid-ex.dsh" 35 33 10 10 65
```



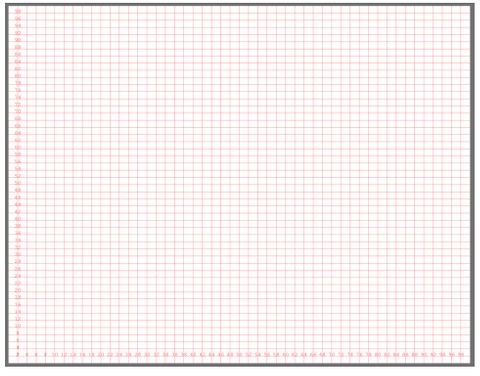

# ruler increment color



ruler



ruler 10



ruler 2 "red"

# Math Functions

| description | keyword | mandatory |
| --- | --- | --- |
| Cosine | `cosine` | `number or expression` |
| Sine | `sine` | `number or expression` |
| Square Root | `sqrt` | `number or expression` |
| Tangent | `tangent` | `number or expression` |

```
a=3.14159265359
b=2*a
x=cosine 4    <------  x = -1
y=cosine a    <------  y = -0.65364
n=cosine a+b  <------  n = -1
```

```
value=cosine  number or expression
```

```
a=10
b=71
x=sin 4          <-----  x = -0.75680
y=sin a          <-----  y = -0.5440
n=sin a+b        <-----  n = -0.62989
```

```
value=sine number or expression
```

```
a=10
b=71
x=sqrt 4
y=sqrt a
n=sqrt a+b
```

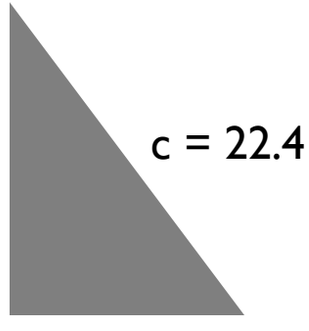
x = 2

y = 3.1622776

n = 9

```
value=sqrt number or expression
```

```
a_squared=10*10
b_squared=20*20
c=sqrt a_squared + b_squared
w=70+10
h=15+20
polygon "70 w 70" "15 15 h"
clabel=format "c = %.1f" c
text clabel 76 25 2
```



```
a=10
b=71
x=tangent 4    <----  x = 1.1578213
y=tangent a    <----  y = 0.6483608
n=tangent a+b  <----  n = -0.8109944
```

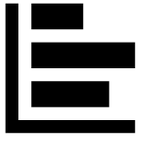
```
value=tangent number or expression
```

# Charts

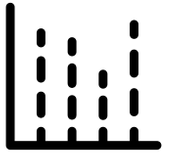
| description | keyword | arguments |
|---|---|---|
| Charts | `dchart` | `options "file" (see next page)` |
| Chart Legends | `legend` | `"text" x y size font color` |

# dcharts types

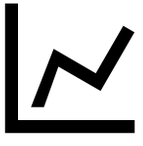
Column



Bar



Dot

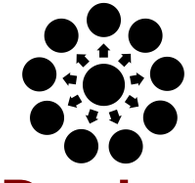


Line

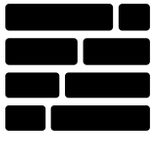
Scatter

Area

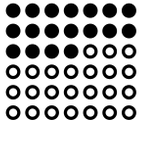
Donut/Pie



Radial



Pmap



Waffle/Lego

Fan

Bowtie

# dchart options: chart types

| option | default | description |
|---|---|---|
| `-bar` | true | bar chart |
| `-wbar` | false | word bar chart |
| `-hbar` | false | horizontal bar chart |
| `-donut` | false | donut chart |
| `-dot` | false | dot chart |
| `-lego` | false | lego chart |
| `-line` | false | line chart |
| `-pgrid` | false | proportional grid |
| `-pmap` | false | proportional map |
| `-bowtie` | false | bowtie chart |
| `-fan` | false | fan chart |
| `-radial` | false | radial chart |
| `-scatter` | false | scatter chart |
| `-slope` | false | slope chart |
| `-vol` | false | volume (area) chart |

# dchart options: elements

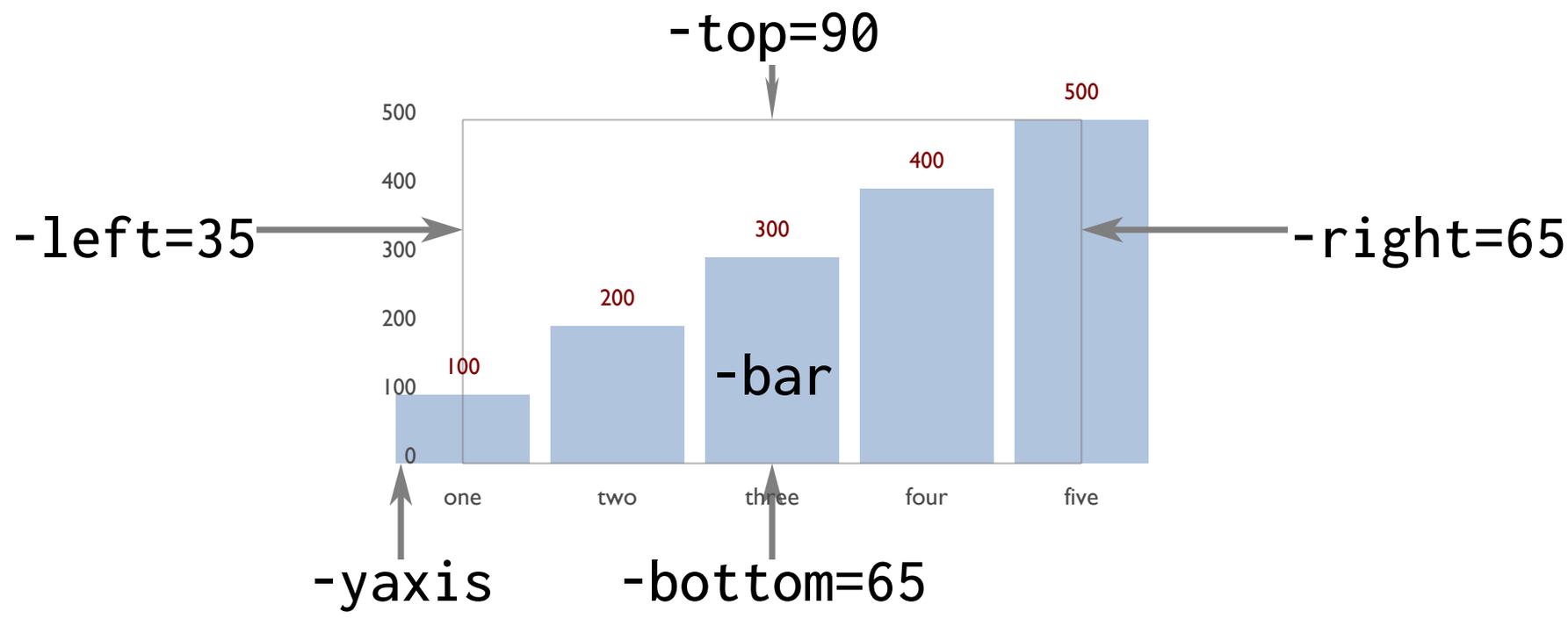
| option | default | description |
|---|---|---|
| -csv | false | read CSV files |
| -frame | false | show a colored frame |
| -fulldeck | true | generate full deck markup |
| -grid | false | show gridlines on the y axis |
| -note | true | show annotations |
| -pct | false | show computed percentage |
| -rline | false | show a regression line |
| -solidpmap | false | show solid pmap colors |
| -spokes | false | show spokes in radial chart |
| -title | true | show the title |
| -val | true | show values |
| -xlast | false | show the last x label |
| -xstagger | false | stagger x axis labels |
| -yaxis | false | show a y axis |
| -chartitle | override title in data | specify the title |
| -datacond | low,high,color | conditional data colors |
| -hline | value,label | label horizontal line at value |
| -valpos | t=top, b=bottom, m=middle | value position |
| -xlabel | default=1, 0 to suppress | x axis label interval |
| -yrange | min,max.step | specify the y axis label range |

# dchart options: measures and attributes

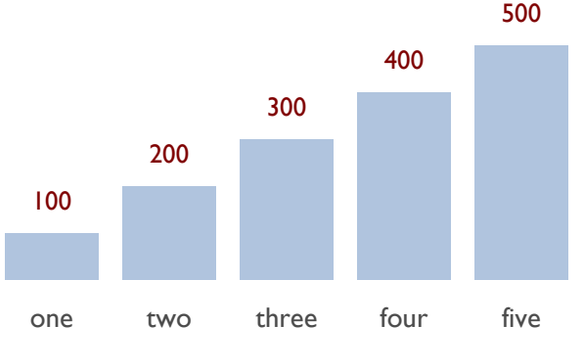
| option | default | description |
|---|---|---|
| -bgcolor | white | background color |
| -barwidth | computed from data size | barwidth |
| -color | lightsteelblue | data color |
| -csvcol | label,label2 | specify csv columns |
| -datafmt | %.1f | data format for values |
| -dmin | false | use data minimum, not zero |
| -framecolor | rgb(127,127,127) | frame color |
| -lcolor | rgb(75,75,75) | label color |
| -linewidth | 0.2 | linewidth |
| -ls | 2.4 | linespacing |
| -noteloc | c=center, r=right, l=left | annotation location |
| -pmlen | 20 | pmap label length |
| -psize | 30 | diameter of the donut |
| -pwidth | 3 | width of the donut or pmap |
| -rlcolor | rgb(127,0,0) | regression line color |
| -textsize | 1.5 | text size |
| -xlabrot | 0 | xlabel rotation (deg.) |
| -vcolor | rgb(127,0,0) | value color |
| -volop | 50 | volume opacity % |

# dchart options: position and scaling

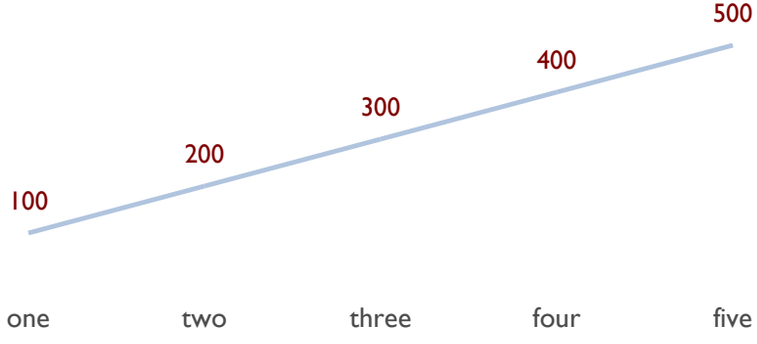
| option | default | description |
|---|---|---|
| `-top` | 80 | top of the chart |
| `-bottom` | 30 | bottom of the chart |
| `-left` | 20 | left margin |
| `-right` | 80 | right margin |
| `-min` | data min | set the minimum data value |
| `-max` | data max | set the maximum data value |
| `-bounds` | "" | set left,right,top,bottom |




# dchart options "file"




```
dchart -left=10 -right=30 -top=35 -bottom=20 "test.d"
```




```
dchart -left=55 -right=85 -top=35 -bottom=20 -bar=f -line "test.d"
```


```
legend  "text" x y fontsize font color
```

■ Item on the chart

■ Thing

```
legend "Item on the chart" 20 30 3 "sans" "red"
```

```
legend "Thing" 70 30 2 "serif" "blue"
```

# Geographic Functions

| description | keyword | mandatory | optional |
|---|---|---|---|
| Geographic Regions | georegion | "file" | color op |
| Geographic Borders | geoborder | "file" | lw color op |
| Text labels | geolabel | "loc" | size font color op |
| Dot markers | geomark | "loc" | size color op |
| Text with markers | geoloc | "loc" | align size font color op |
| Place images | geoimage | "loc" w h | |
| Lines between points | geopath | "p1" "p2" | lw color op |
| Arcs between points | geoarc | "p1" "p2" | lw color op |
| Lines between points | geopathfile | "file" | lw color op |

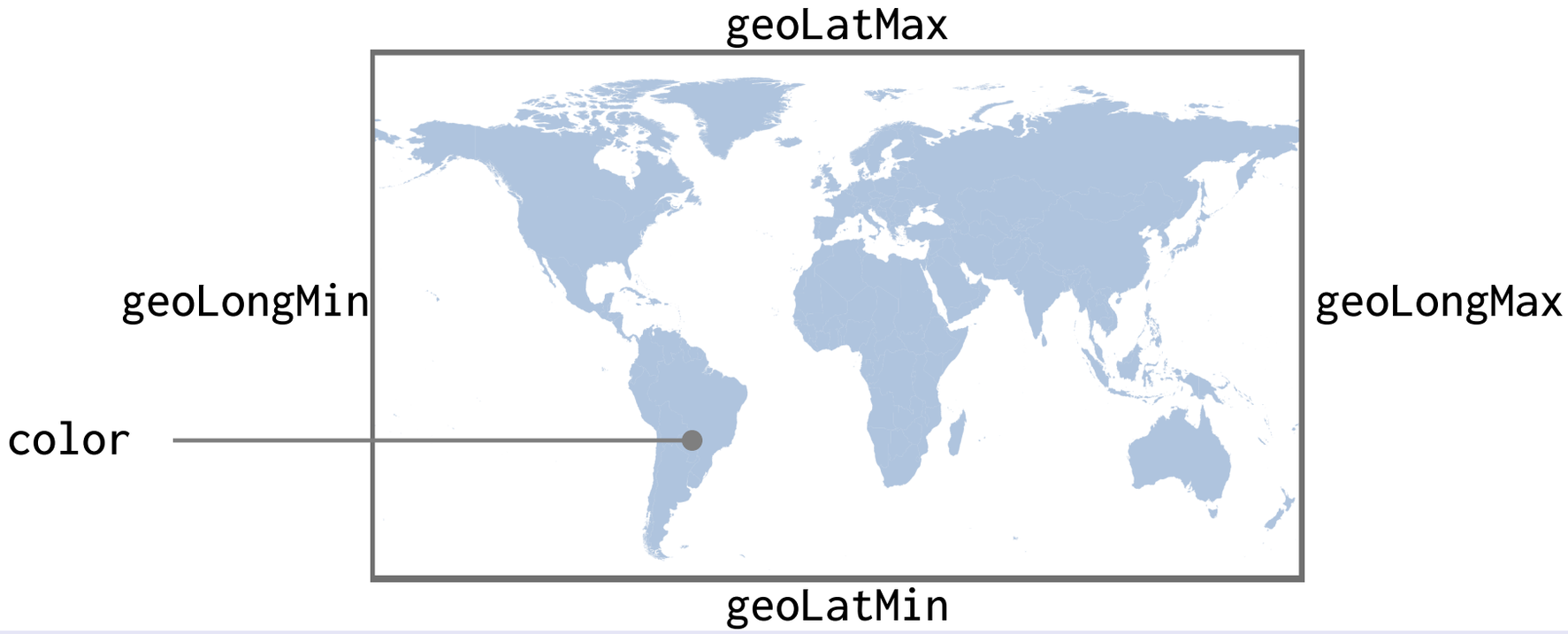
"file" refer to Shapefile (.shp), GeoJSON (.json or .geojson) or Keyhole Markup Language (.kml) files from opendatasoft

geoLatMin, geoLatMax, geoLongMin, geoLongMax variables set the latitude and longitude boundaries

geoXmin, geoXmax, geoYmin, geoYmax variables define the canvas boundaries for geographic functions

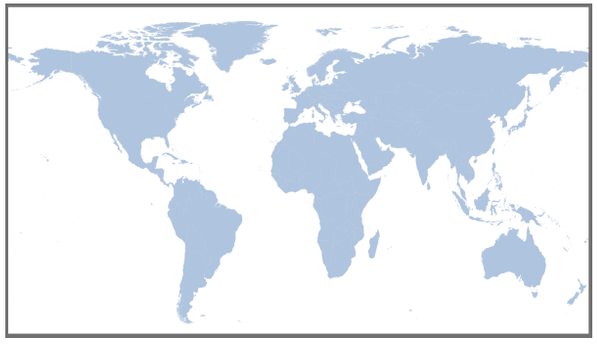
"loc" refers to a geo URI string ("geo:lat,long"), or a filename containing multiple locations

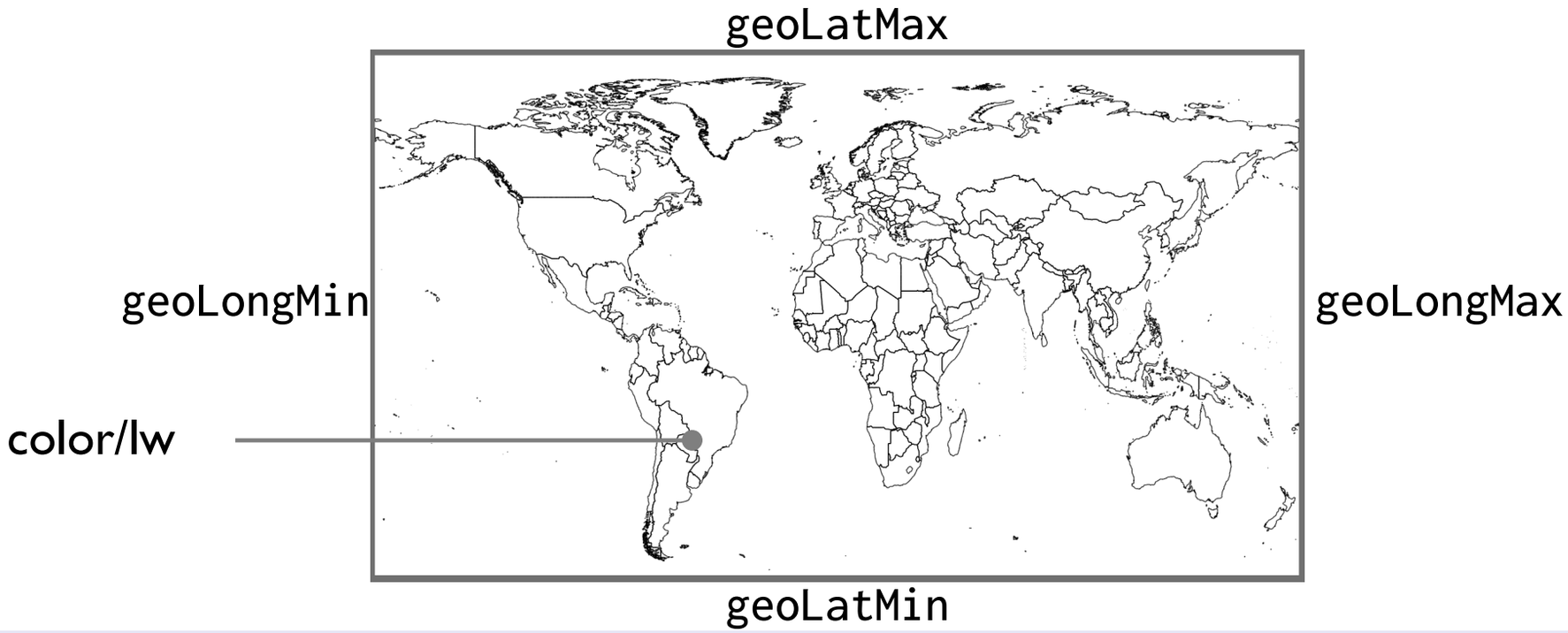
"p1" and "p2" are geo URIs




```
georegion "file"       color op
```

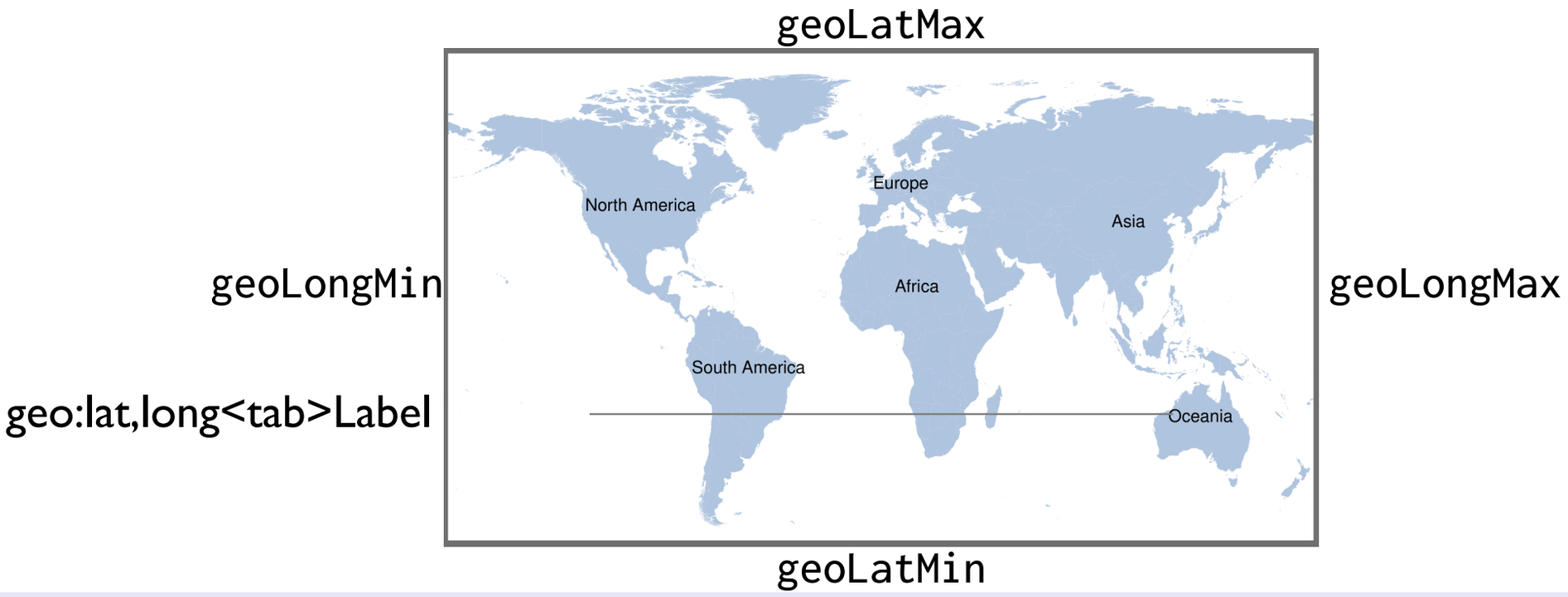
```
geoLatMin=0-60
geoLatMax=90
geoLongMin=0-180
geoLongMax=180

georegion "world.kml"  "white"
```






```
geoborder "file"        lw color
```

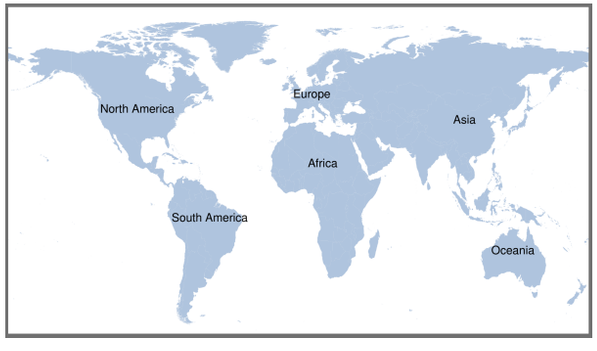
```
geoLatMin=0-60
geoLatMax=90
geoLongMin=0-180
geoLongMax=180

geoborder "world.kml"  0.1 "black"
```



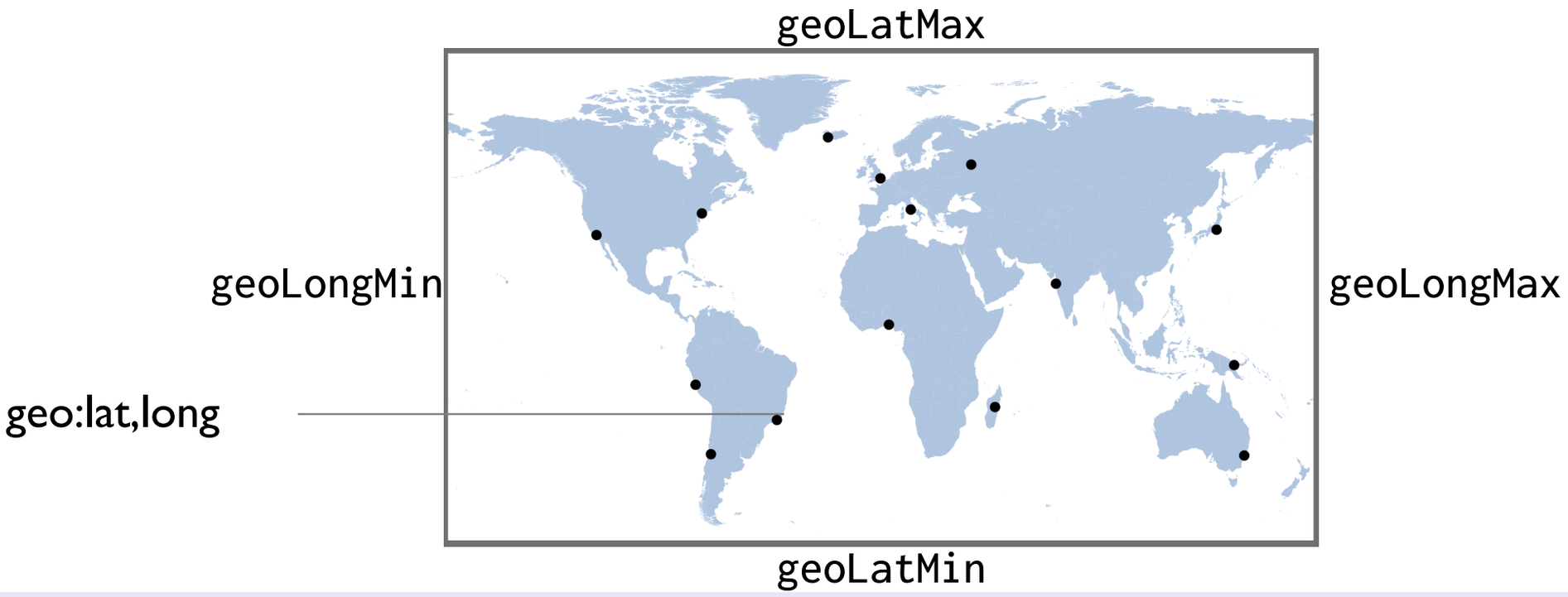

```
geolabel  "loc"  size color op
```

```
geoLatMin=0-60
geoLatMax=90
geoLongMin=0-180
geoLongMax=180
```

```
geo:-10,-55    South America
geo:40,-100    North America
geo:46.8,8.3    Europe
geo:15,15    Africa
geo:35,103    Asia
geo:-25,133    Oceania
```



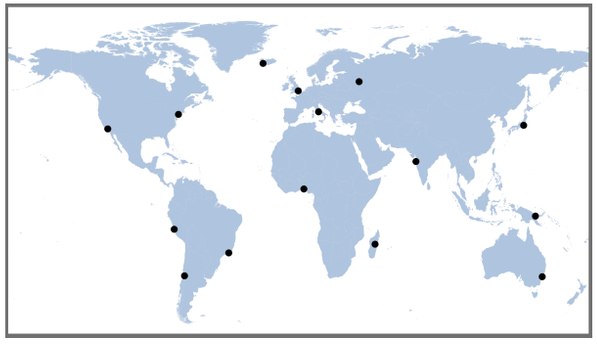

```
georegion "world.kml"  "white"
geolabel "continents.d"  2 "sans" "black"
```




```
geomark "loc"    size color op
```

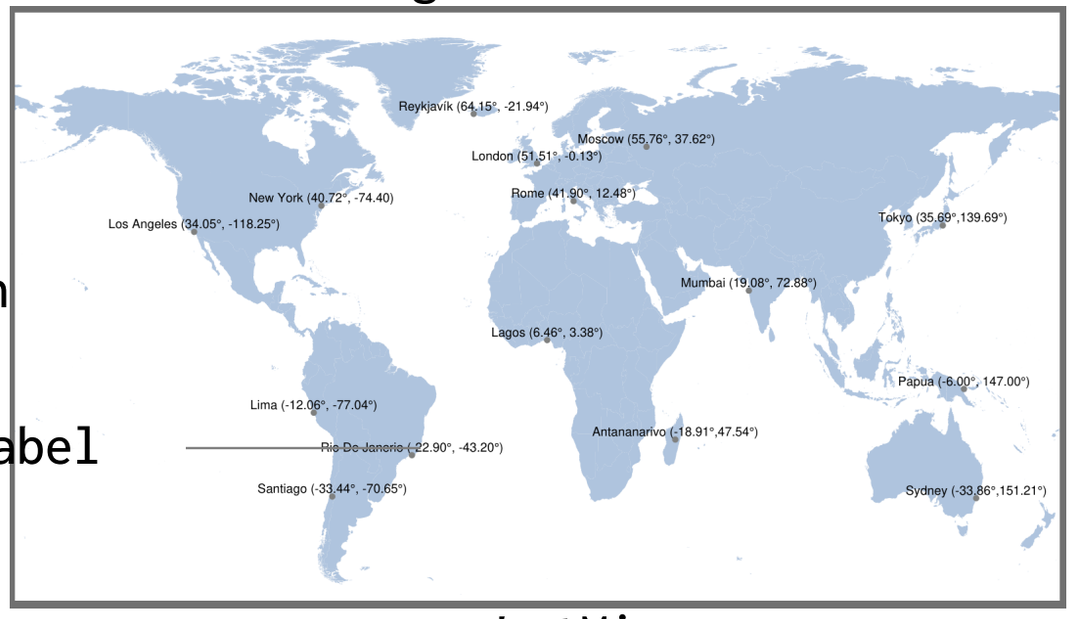
```
geoLatMin=0-60
geoLatMax=90
geoLongMin=0-180
geoLongMax=180
```

```
geo:41.8967,12.4822000    Rome (41.90°, 12.48°)
geo:-18.91368,47.53610    Antananarivo (-18.91°,47.54°)
geo:-33.8559,151.20670    Sydney (-33.86°,151.21°)
geo:40.7167,-74.400000    New York (40.72°, -74.40)
...
geo:34.0500,-118.25000    Los Angeles (34.05°, -118.25°)
```



```
georegion "world.kml"   "white"
geolabel "cities.d"     2 "sans" "black"
```

geoLatMax




geoLongMin

geoLongMax

geo:lat,long<tab>Label

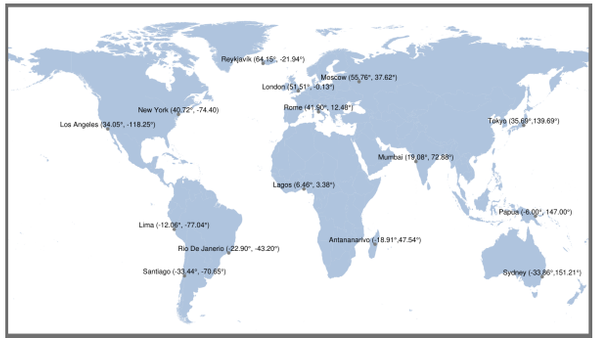
Label Position (pos):
"c" Center above
"b" Begin Aligned
"e" End Aligned
"u" Center below

geoLatMin

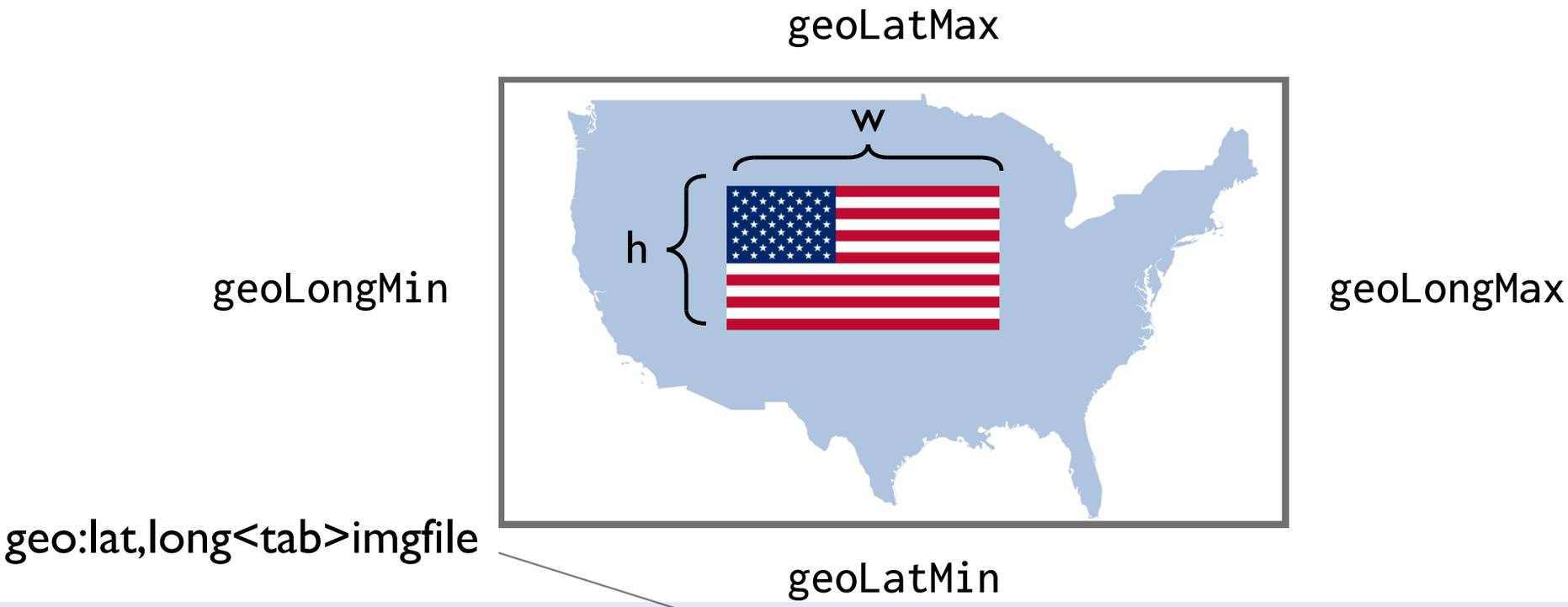
```
geoloc "loc"  pos size font color op
```

```
geoLatMin=0-60
geoLatMax=90
geoLongMin=0-180
geoLongMax=180
```

```
geo:41.8967,12.4822000    Rome (41.90°, 12.48°)
geo:-18.91368,47.53610    Antananarivo (-18.91°,47.54°)
geo:-33.8559,151.20670    Sydney (-33.86°,151.21°)
geo:40.7167,-74.400000    New York (40.72°, -74.40)
...
geo:34.0500,-118.25000    Los Angeles (34.05°, -118.25°)
```

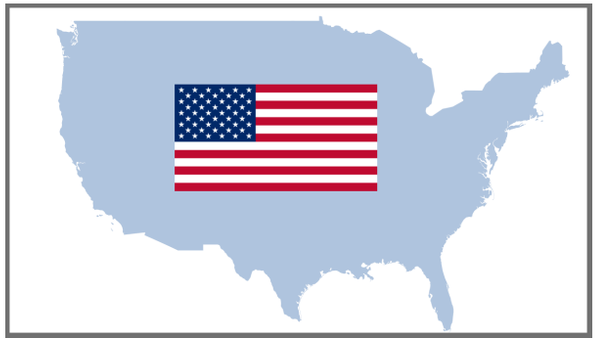


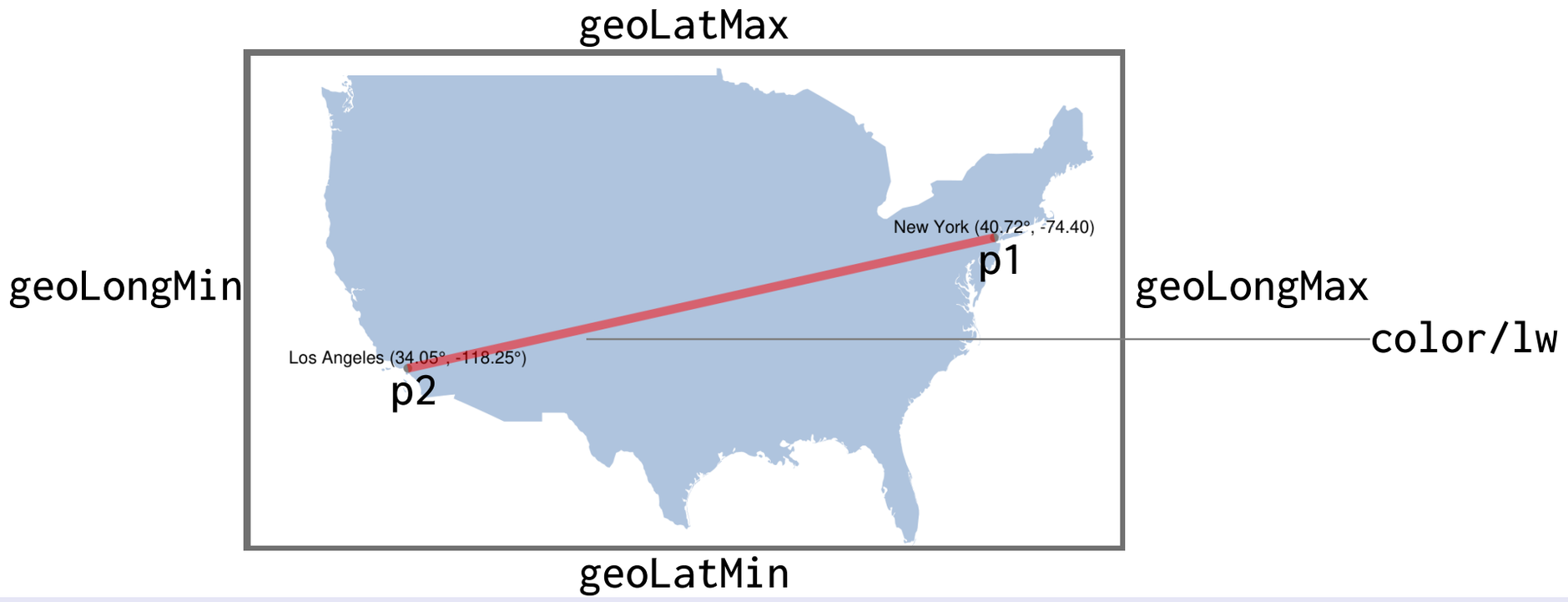

```
georegion "world.kml"  "white"
geoloc "cities.d"      "c" ts
```




```
geoimage  "loc" w h
```

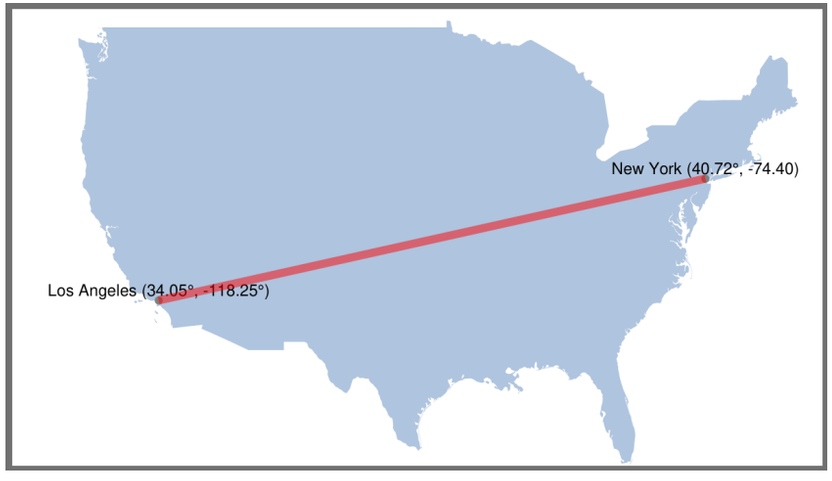
```
geoLatMin=25
geoLatMax=50
geoLongMin=0-130
geoLongMax=0-65

georegion  "usa.kml"  "lightsteelblue"
geoimage   "geo:40,-100    usa.png"  35 0
```

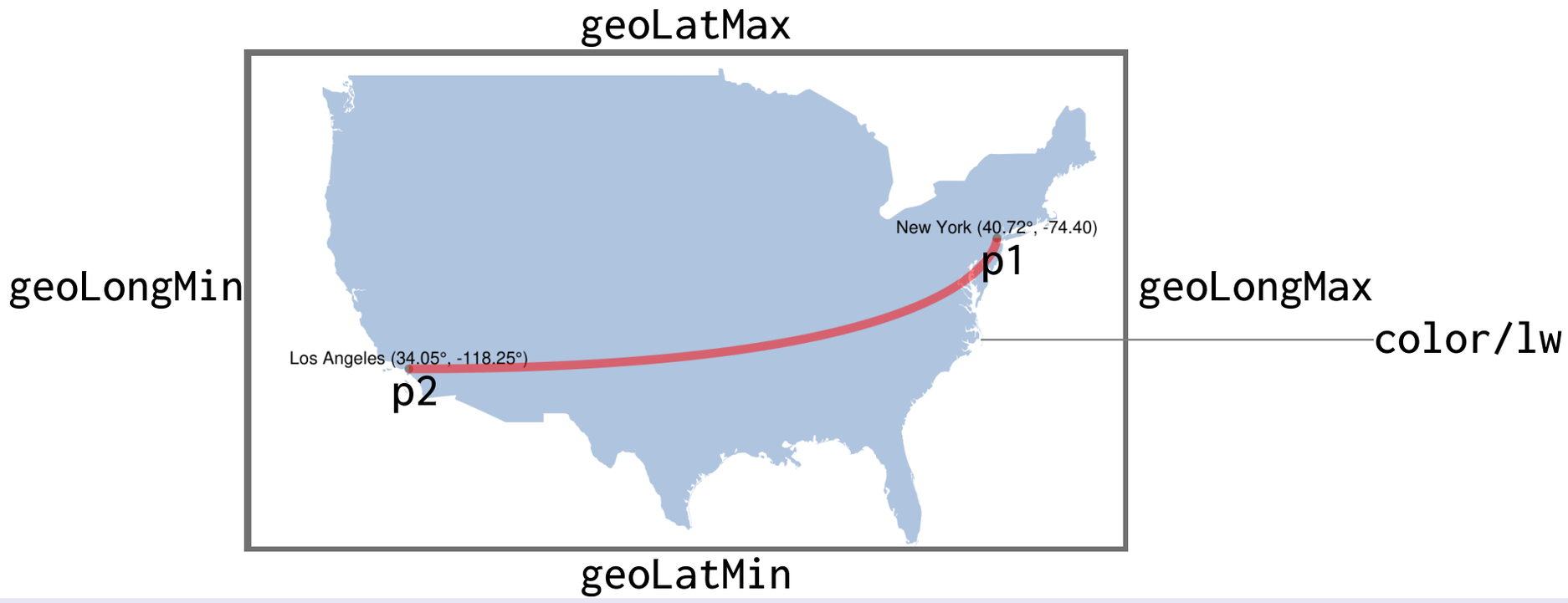





```
geopath  "p1"  "p2"   lw   lw color op
```

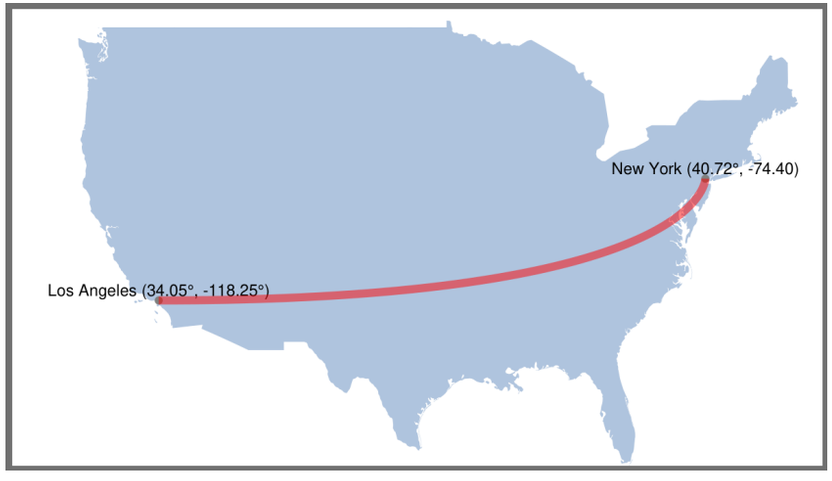
```
geoLatMin=25
geoLatMax=50
geoLongMin=0-130
geoLongMax=0-65
nyc="geo:+40.7167,-74.4000    New York (40.72°, -74.40)"
los="geo:+34.05000,-118.250    Los Angeles (34.05°, -118.25°)"

georegion "usa.kml"  "lightsteelblue"
geoloc    nyc  "c" 2
geoloc    los  "c" 2
geopath    nyc los  1 "red" 50
```

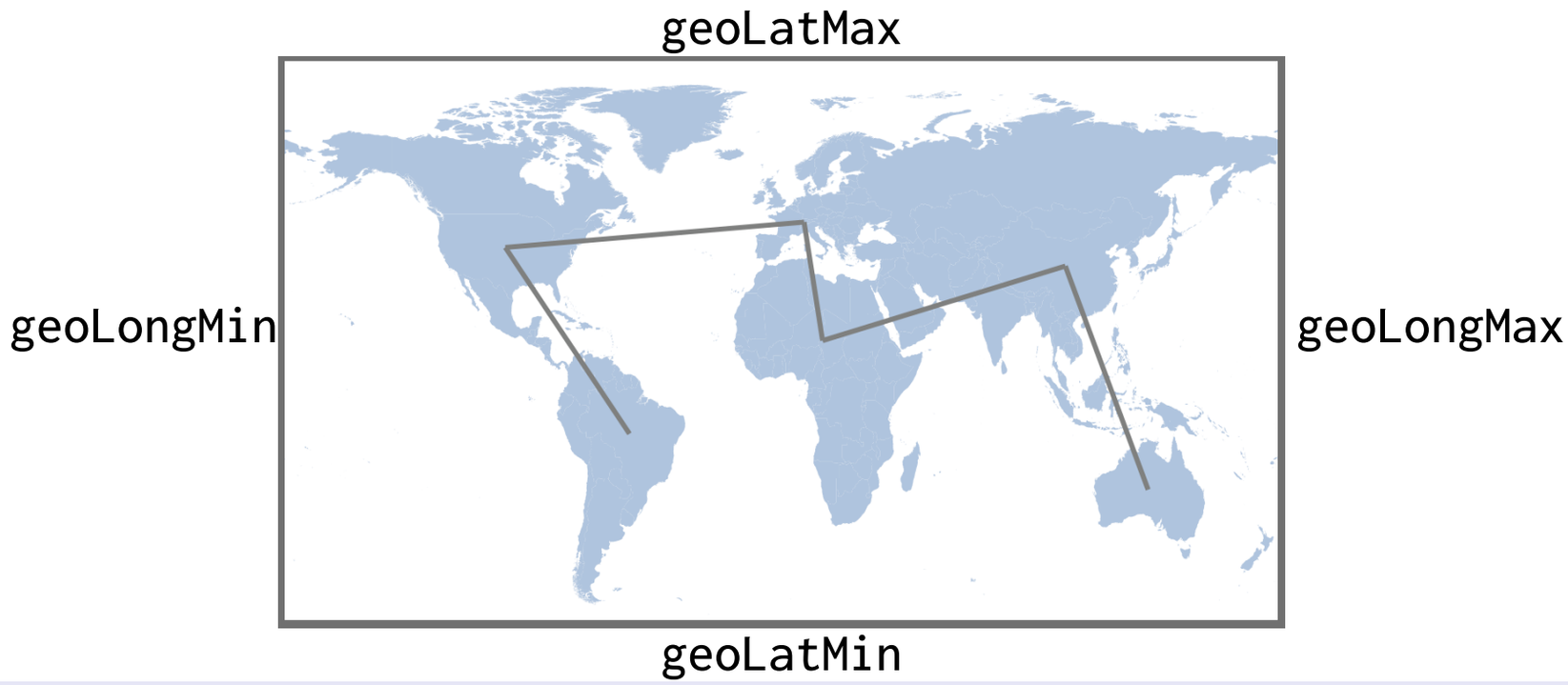





```
geoarc   "p1" "p2"   lw   color op
```

```
geoLatMin=25
geolatmax=50
geoLongMin=0-130
geoLongMax=0-65
nyc="geo:+40.7167,-74.4000    New York (40.72°, -74.40)"
los="geo:+34.05000,-118.250    Los Angeles (34.05°, -118.25°)"

georegion "usa.kml" "lightsteelblue"
geoloc    nyc "c" 2
geoloc    los "c" 2
geoarc    nyc los 1 "red" 50
```






```
geopathfile "file"  lw color op
```

```
geoLatMin=25
geoLatMax=50
geoLongMin=0-130
geoLongMax=0-65
```

```
geo:40.712778,-74.006111    New York
geo:39.952778,-75.163611    Philadelphia
geo:39.768611,-86.158056    Indianapolis
geo:38.627222,-90.197778    St. Louis
geo:35.468611,-97.521389    Oklahoma City
geo:35.084444,-106.650278    Albuquerque
geo:36.167222,-115.148611    Las Vegas
geo:34.050000,-118.250000    Los Angeles
```


```
georegion   "usa.kml"      "lightsteelblue"
geoloc      "roadtrip.d"   "c" 2
geopathfile "roadtrip.d"   0.5 "red" 25
```