

# decksh reference



# Keywords

## Structure Text

deck  
edeck  
slide  
eslide  
canvas  
def  
edef  
func  
grid  
include

## Lists

list  
blist  
nlist  
clist  
li  
elist

## Graphics

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

## Braces Arrows

lbrace  
rbrace  
ubrace  
dbrace  
lbracket  
rbracket  
dbracket  
ubracket  
arrow  
rcarrow  
lcarrow  
ucarrow  
dcarrow

## Images

image  
cimage

## Charts

dchart  
legend

## Loop

for  
efor

## Assignments

polar  
polarx  
polary  
area  
format  
random  
sqrt  
vmap

## Data

data  
edata

# Keywords and arguments

keyword	arguments						
	mandatory				optional		
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

# Structure

```
// This is a comment
deck
    canvas 1920 1080
    variables {
        x=20      // define x
        y=80
    }
    slide 1 {
        slide
        text "first" x y 2
    }
    eslide
    slide 2 {
        slide "black" "white"
        include "file.dsh"
    }
    eslide
edeck
```

← comment

← canvas size hint (width height)

← inline comment

deck {

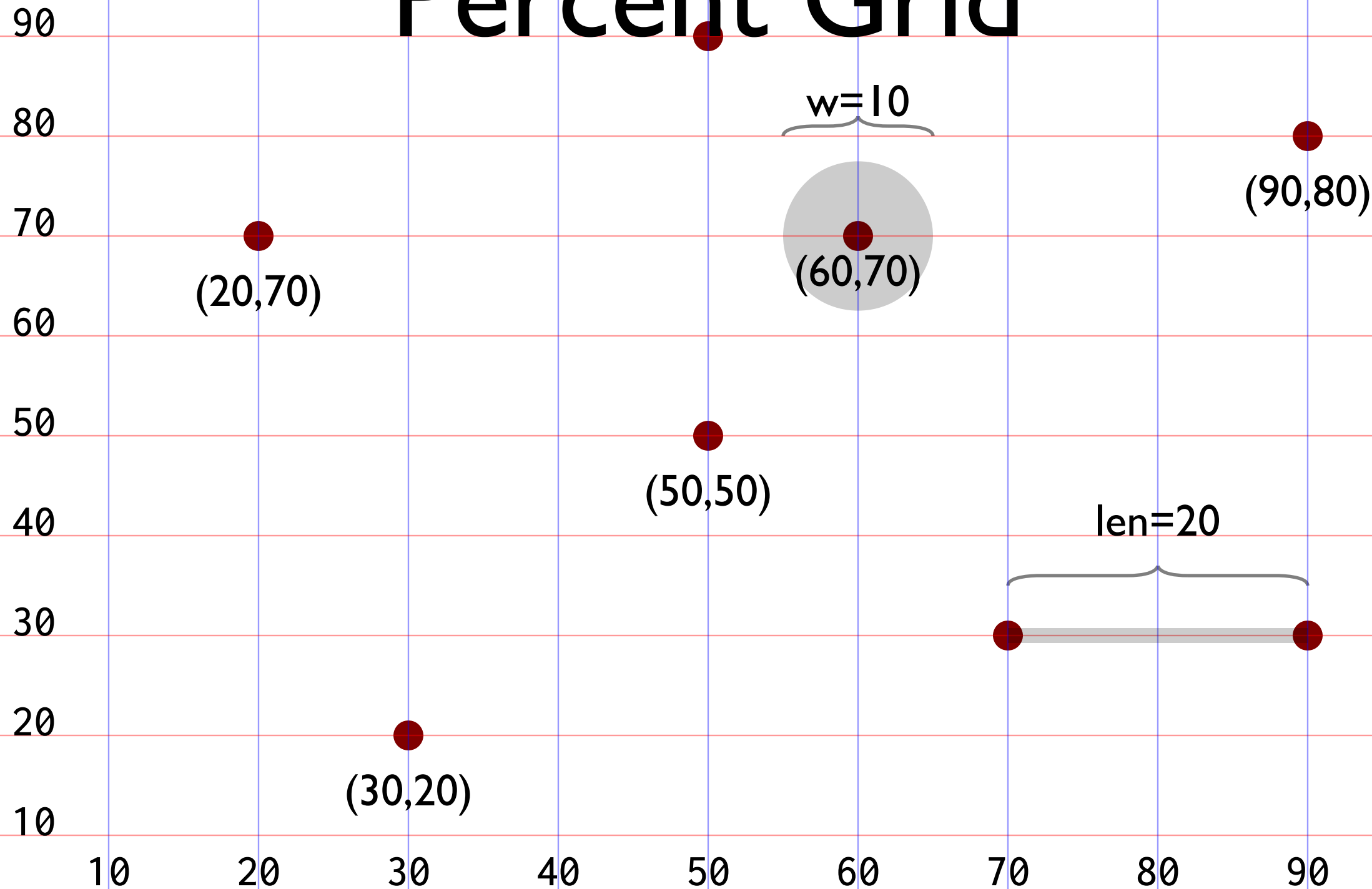
variables {

slide 1 {

slide 2 {

```
ctext "hello, world" 50 25 10
circle 50 0 100 "blue"
for x=20 80 10
    circle x 75 2
efor
```

# Percent Grid



# 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

# Binary operators

$x = a + b$

addition

$x = a - b$

subtraction

$x = a * b$

multiplication

$x = a / b$

division

# 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=polar y cx cy r theta`

polar coordinate (y)

`v=random v1 v2`

random number

`v=vmap data v1 v2 v3 v4`

range map

`v=area expr`

area

`v=sqrt expr`

square root




# Variables and Assignments

Using variables	x=10	// assign 10 to x
	y=80.45	// assign 80.45 to y
	size=x/5	// x divided by 5 to size
	s="Hello"	// s has the string value 'Hello'
	text s x y size	// Use the variables

Binary operations	a=100+200.7	// addition
	a=x-10	// subtraction
	a=x*10	// multiplication
	a=x/y	// division

Assignment operations	x+=10	// increase x by 10
	x-=10	// decrease x by 10
	x*=10	// multiply x by 10
	x/=y	// divide x by value of y

# Colors, fonts, opacity, gradient

Colors	Fonts		Opacity (0-100)	
"steelblue"	"sans"	Sans Serif	100	
"#4682b4"	"serif"	Serif	50	
"rgb(70,130,180)"	"mono"	Monospace	10	
	"symbol"	❁❄❅❆❇		

"red/blue/90" 

(applies to rect and square only)

# Color Index

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

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

## Color Index (2)

name	hex	RGB	name	hex	RGB
 darkred	#8b0000	rgb(139,0,0)	 fuchsia	#ff00ff	rgb(255,0,255)
 darksalmon	#e9967a	rgb(233,150,122)	 gainsboro	#dcdcdc	rgb(220,220,220)
 darkseagreen	#8fbc8f	rgb(143,188,143)	 ghostwhite	#f8f8ff	rgb(248,248,255)
 darkslateblue	#483d8b	rgb(72,61,139)	 gold	#ffd700	rgb(255,215,0)
 darkslategray	#2f4f4f	rgb(47,79,79)	 goldenrod	#daa520	rgb(218,165,32)
 darkslategrey	#2f4f4f	rgb(47,79,79)	 gray	#808080	rgb(128,128,128)
 darkturquoise	#00ced1	rgb(0,206,209)	 green	#008000	rgb(0,128,0)
 darkviolet	#9400d3	rgb(148,0,211)	 greenyellow	#adff2f	rgb(173,255,47)
 deeppink	#ff1493	rgb(255,20,147)	 grey	#808080	rgb(128,128,128)
 deepskyblue	#00bfff	rgb(0,191,255)	 honeydew	#f0fff0	rgb(240,255,240)
 dimgray	#696969	rgb(105,105,105)	 hotpink	#ff69b4	rgb(255,105,180)
 dimgrey	#696969	rgb(105,105,105)	 indianred	#cd5c5c	rgb(205,92,92)
 dodgerblue	#1e90ff	rgb(30,144,255)	 indigo	#4b0082	rgb(75,0,130)
 firebrick	#b22222	rgb(178,34,34)	 ivory	#fffff0	rgb(255,255,240)
 floralwhite	#fffaf0	rgb(255,250,240)	 khaki	#f0e68c	rgb(240,230,140)
 forestgreen	#228b22	rgb(34,139,34)	 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	#ffffac	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	name	hex	RGB
 midnightblue	#191970	rgb(25,25,112)	 papayawhip	#ffefd5	rgb(255,239,213)
 mintcream	#f5fffa	rgb(245,255,250)	 peachpuff	#ffdab9	rgb(255,218,185)
 mistyrose	#ffe4e1	rgb(255,228,225)	 peru	#cd853f	rgb(205,133,63)
 moccasin	#ffe4b5	rgb(255,228,181)	 pink	#ffc0cb	rgb(255,192,203)
 navajowhite	#ffdead	rgb(255,222,173)	 plum	#dda0dd	rgb(221,160,221)
 navy	#000080	rgb(0,0,128)	 powderblue	#b0e0e6	rgb(176,224,230)
 oldlace	#fdf5e6	rgb(253,245,230)	 purple	#800080	rgb(128,0,128)
 olive	#808000	rgb(128,128,0)	 red	#ff0000	rgb(255,0,0)
 olivedrab	#6b8e23	rgb(107,142,35)	 rosybrown	#bc8f8f	rgb(188,143,143)
 orange	#ffa500	rgb(255,165,0)	 royalblue	#4169e1	rgb(65,105,225)
 orangered	#ff4500	rgb(255,69,0)	 saddlebrown	#8b4513	rgb(139,69,19)
 orchid	#da70d6	rgb(218,112,214)	 salmon	#fa8072	rgb(250,128,114)
 palegoldenrod	#eee8aa	rgb(238,232,170)	 sandybrown	#f4a460	rgb(244,164,96)
 palegreen	#98fb98	rgb(152,251,152)	 seagreen	#2e8b57	rgb(46,139,87)
 paleturquoise	#afeeee	rgb(175,238,238)	 seashell	#fff5ee	rgb(255,245,238)
 palevioletred	#db7093	rgb(219,112,147)	 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	#ffaafa	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)

name	hex	RGB
 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	#ffaafa	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)




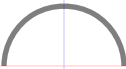
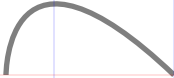


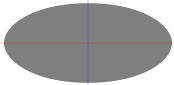
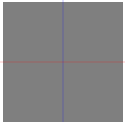
















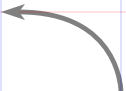
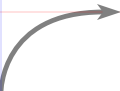
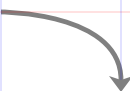

name	hex	RGB
 mediumblue	#0000cd	rgb(0,0,205)
 mediumslateblue	#7b68ee	rgb(123,104,238)
 mediumturquoise	#48d1cc	rgb(72,209,204)
 midnightblue	#191970	rgb(25,25,112)
 navy	#000080	rgb(0,0,128)
 paleturquoise	#afeeee	rgb(175,238,238)
 powderblue	#b0e0e6	rgb(176,224,230)
 royalblue	#4169e1	rgb(65,105,225)
 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	#ffffac	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)

text	ctext	etext	rtext	arctext	textbox	textfile	textcode	
left	center	right	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.	<pre>import "fmt" func main() {     fmt.Println("hello, world") }</pre>	
line	hline	vline	arc	curve	circle	acircle	ellipse	
								
square	rect	rrect	pill	polygon	star	image	cimage	
								
lbrace	rbrace	ubrace	dbrace	lbracket	rbracket	ubracket	dbracket	
								
arrow	lcarrow	rcarrow	dcarrow	ucarrow	list	blist	nlist	clist
					one two three	<ul style="list-style-type: none"><li>one</li><li>two</li><li>three</li></ul>	<ol style="list-style-type: none"><li>one</li><li>two</li><li>three</li></ol>	first second item thrid



# Textual Elements

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

hello, world

(x,y)

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

abc

text "abc" 20 20 4

abc

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

hello, world

(x,y)

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

abc

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

abc

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

hello, world.

(x,y)

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

abc

`etext "abc" 20 20 4`

abc

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

hello, world

(x,y)

**rttext** "... " x y angle fontsize font color op link

abc

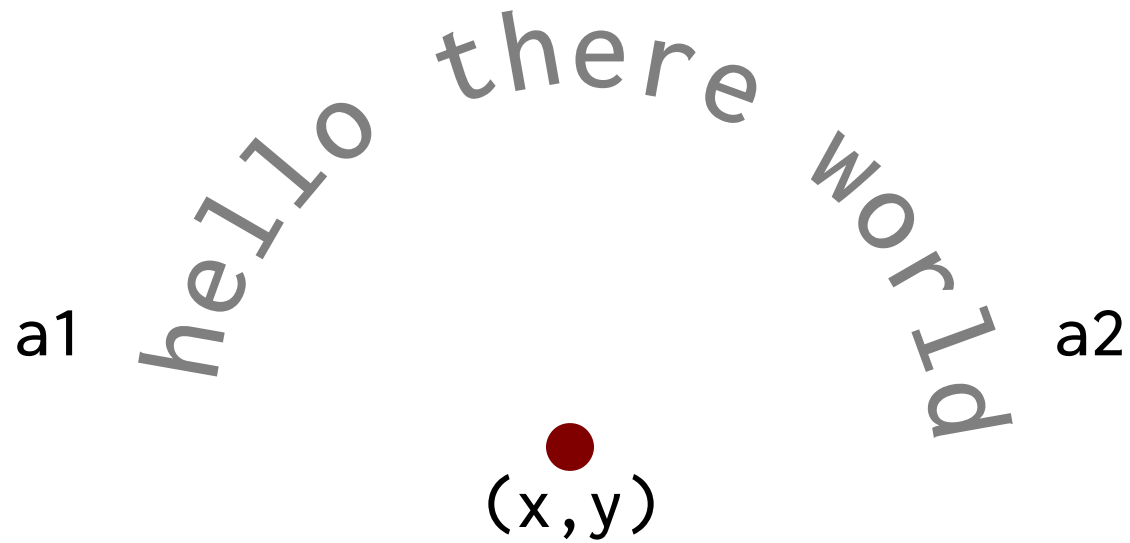
ctext 20 20 30 3

abc

ctext 50 20 90 5

abc

ctext 80 20 270 4 "sans" "maroon"



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

What is up

This is curvy

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

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

(x, y)  “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** "... " 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"

(x,y) This is the contents  
of a file. it has lines of text.  
Reading is fundamental.

`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("hello, world")
}
```

`textfile "example.txt" 10 35 2`

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



(x,y)

W

```
import "fmt"
func main() {
    fmt.Println("hello, world")
}
```

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

```
import "fmt"
func main() {
    fmt.Println("hello, world")
}
```

textcode "hw.go" 10 35 25 1.0

```
import "fmt"
func main() {
    fmt.Println("hello, world")
}
```

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

# Graphical Elements

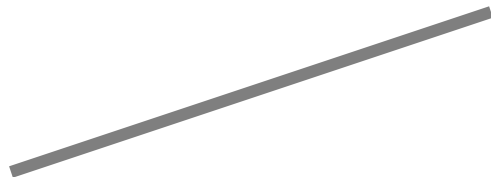
description	keyword	mandatory	optional
Line	<b>line</b>	x1 y1 x2 y2 lw	color op
Horizontal line	<b>hline</b>	x y w	lw color op
Vertical line	<b>vline</b>	x y h	lw color op
Elliptical arc	<b>arc</b>	x y w h a1 a2	lw color op
Quadratic Bezier	<b>curve</b>	bx by cx cy ex ey	lw color op
Circle	<b>circle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

lw {   
(x1,y1) (x2,y2)

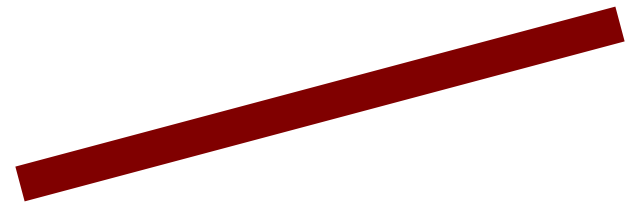
line x1 y1 x2 y2 lw color op



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`



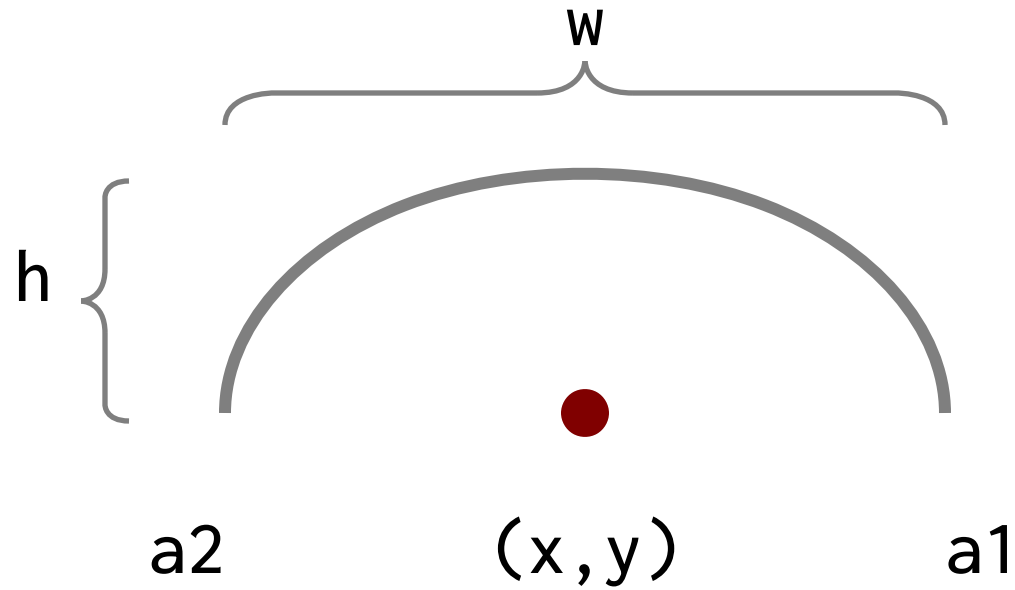
`vline 20 20 15`



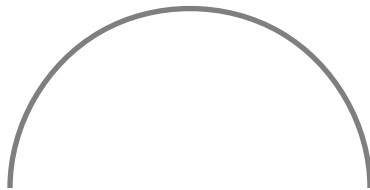
`vline 50 20 15 2`



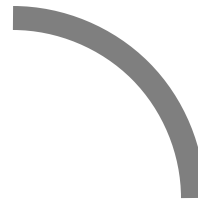
`vline 80 20 15 10 "maroon" 20`



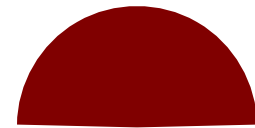
`arc x y w h a1 a2 lw color op`



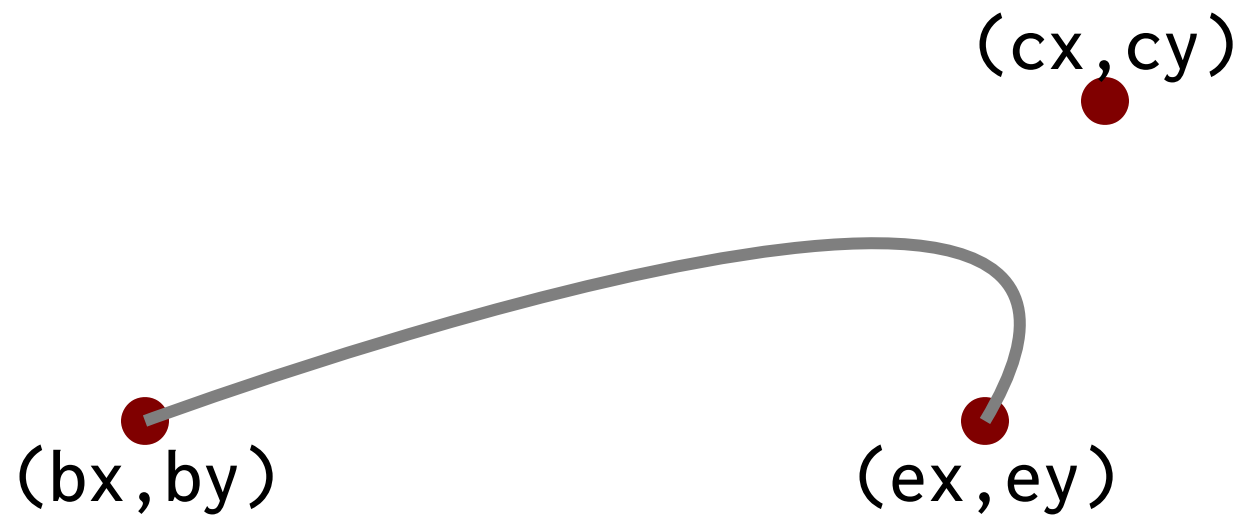
`arc 20 20 15 15 0 180`



`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`

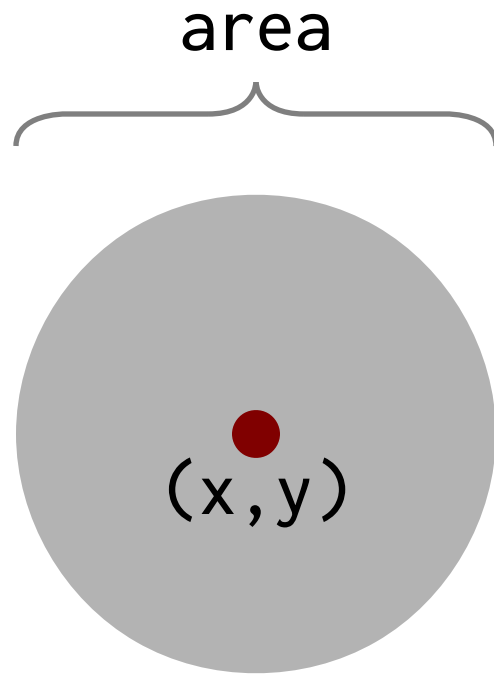


`circle 50 20 10 "maroon"`



`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"`



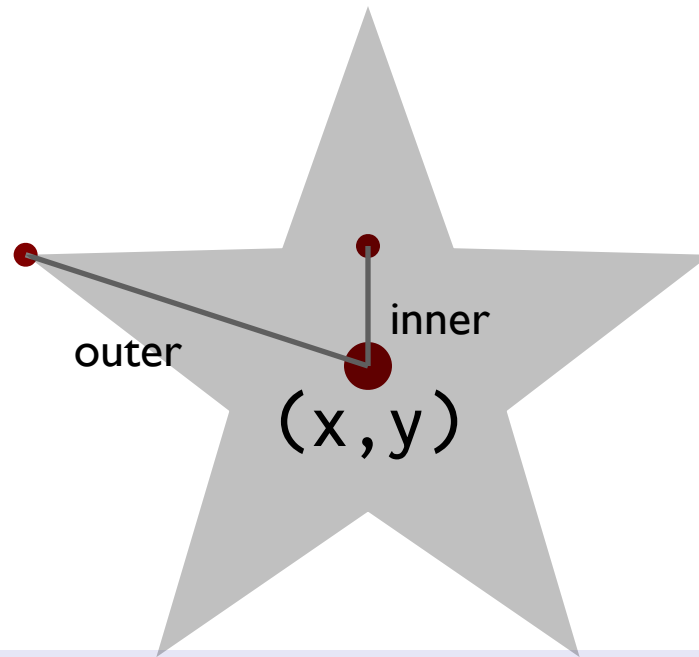
`polygon x1 x2...xn" "y1 y2...yn color op`



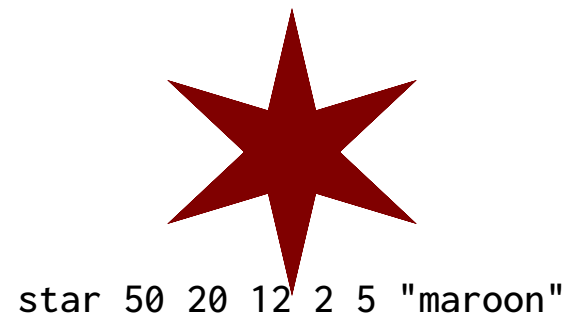
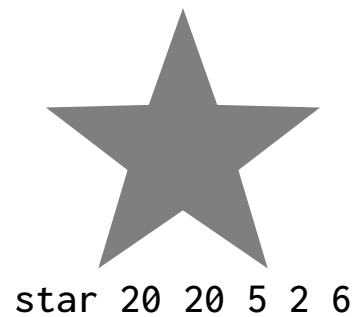
`polygon "10 25 20" "20 30 20"`



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



`star x y sides inner outer color op`





# Images

description	keyword	mandatory	optional
Image	<code>image</code>	<code>"file" x y w h</code>	<code>scale "link"</code>
Captioned image	<code>cimage</code>	<code>"file" "caption" x y w h</code>	<code>scale "link"</code>

Note: the scale value is a percentage from 1-100, and link is a URL

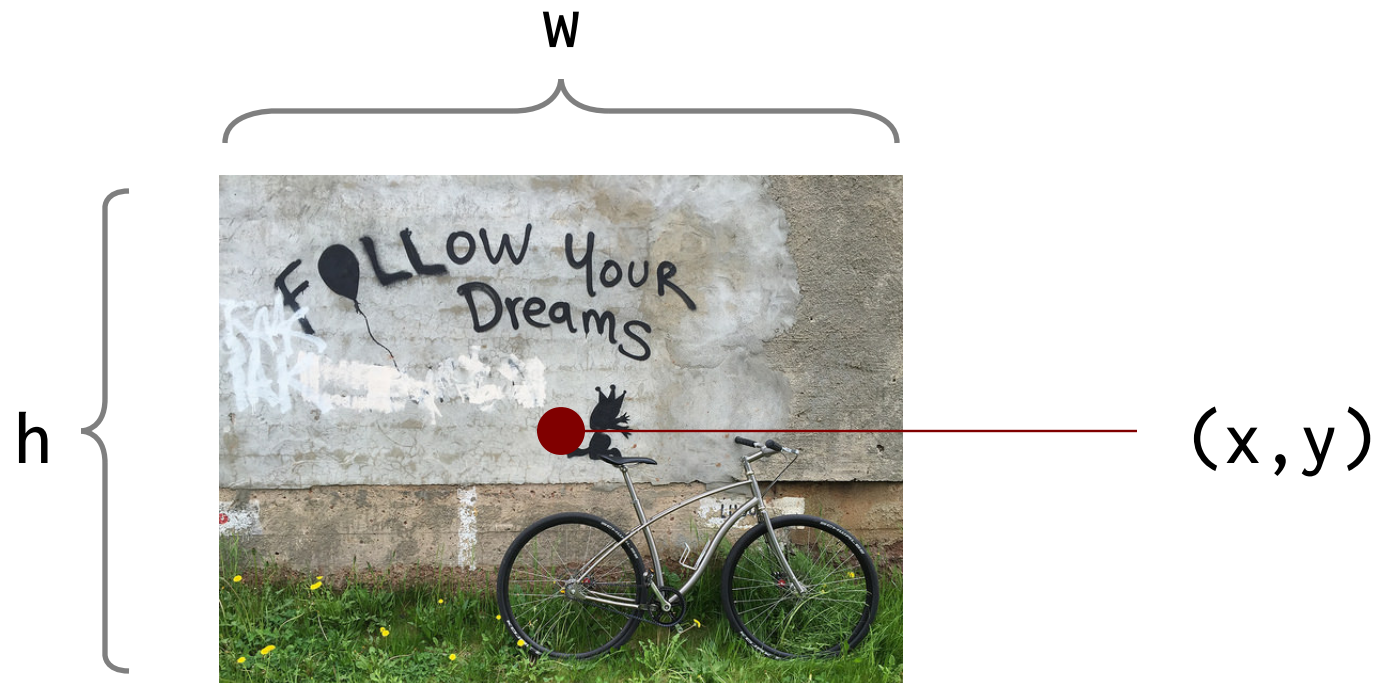


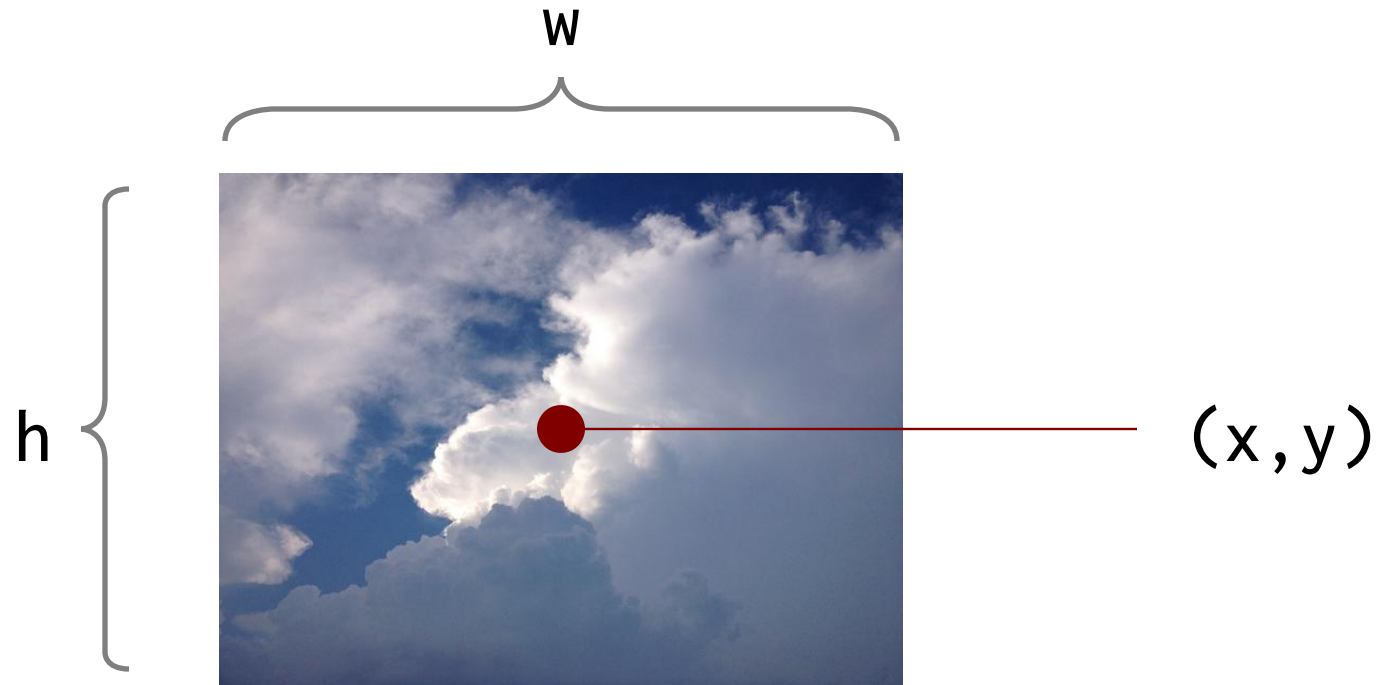
image x y w h scale link



image "follow.jpg" 20 25 640 480 10

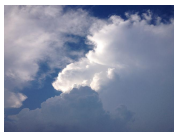


image "follow.jpg" 75 25 640 480 30



sky

`cimage 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	<code>list</code>	<code>x y fontsize</code>	<code>font color op spacing</code>
Bullet list	<code>blist</code>	<code>x y fontsize</code>	<code>font color op spacing</code>
Numbered list	<code>nlist</code>	<code>x y fontsize</code>	<code>font color op spacing</code>
Centered list	<code>clist</code>	<code>x y fontsize</code>	<code>font color op spacing</code>

```
list x y fs
(x,y) li "first"
      li "second"
      li "third"
elist
```

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

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

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

```
      blist x y fs
(x,y)  li "first"
        li "second"
        li "third"
      elist
```

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

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

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

```
nlist x y fs
(x,y) li "first"
      li "second"
      li "third"
elist
```

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

```
nlist 20 30 2.5 1. one
      li "one"
      li "two" 2. two
      li "three" 3. three
elist
```

```
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 fs
(x,y) li "first"
      li "second"
      li "third"
elist
```

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

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

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



# Arrows

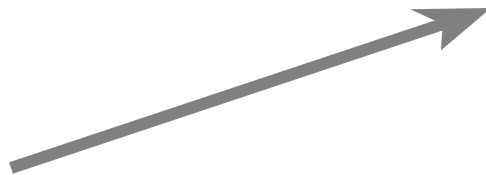
description	keyword	mandatory	optional
Straight	<b>arrow</b>	x1 y1 x2 y2	lw aw ah color op
Left curved	<b>lcarrow</b>	bx by cx cy ex ey	lw aw ah color op
Right curved	<b>rcarrow</b>	bx by cx cy ex ey	lw aw ah color op
Up curved	<b>ucarrow</b>	bx by cx cy ex ey	lw aw ah color op
Down curved	<b>dcarrow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



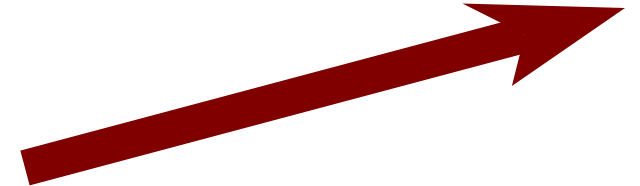
`arrow` `x1` `y1` `x2` `y2` `lw` `aw` `ah` `color` `op`



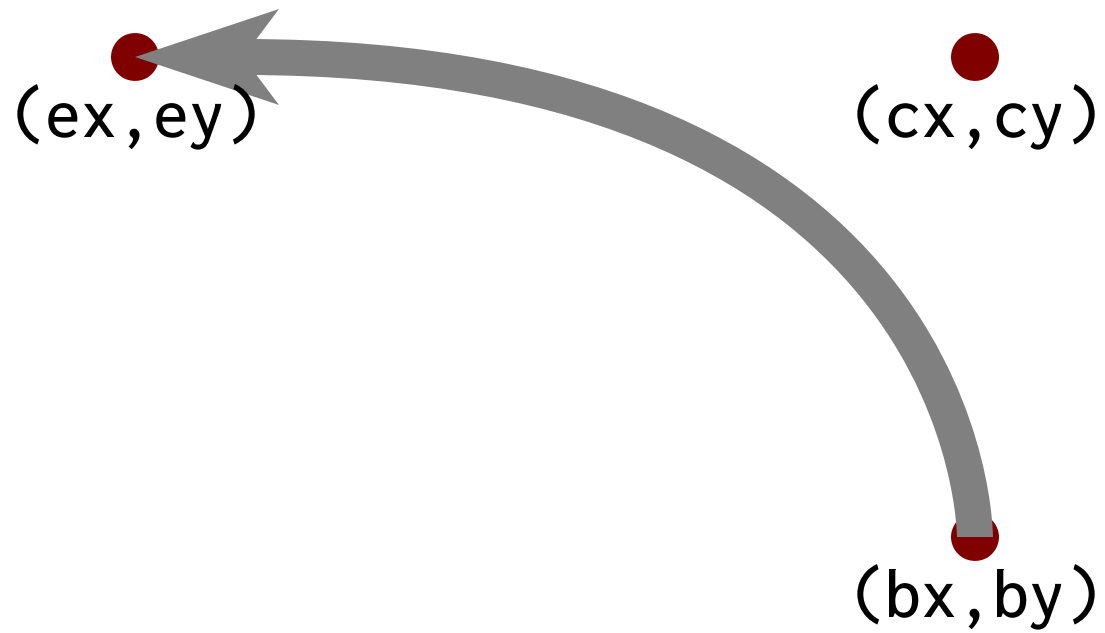
`arrow 10 20 30 20`



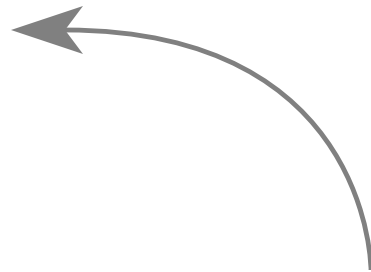
`arrow 40 20 60 30 0.5`



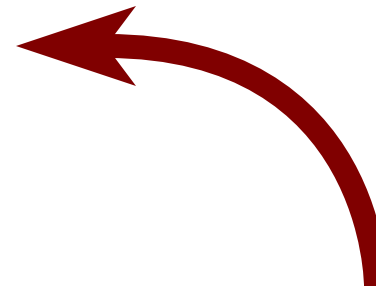
`arrow 70 20 95 30 1.5 6 6 "maroon"`



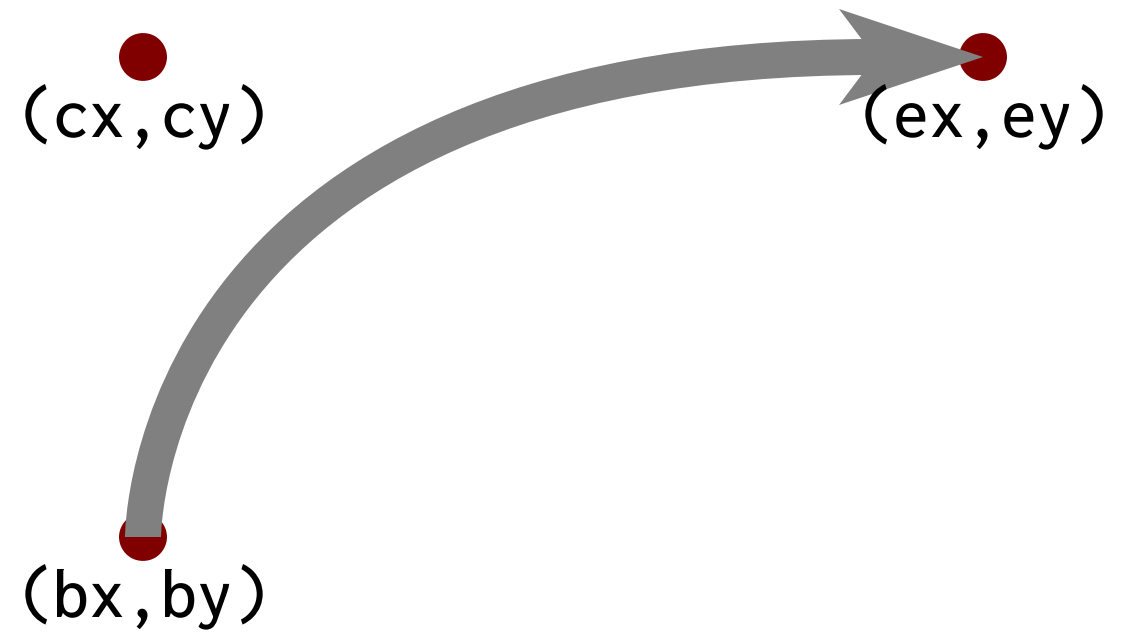
`lcarrow` `bx by cx cy ex ey lw aw ah color op`



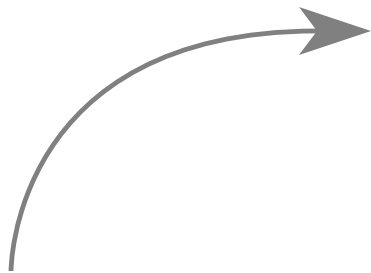
`lcarrow 30 20 30 35 15 35`



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



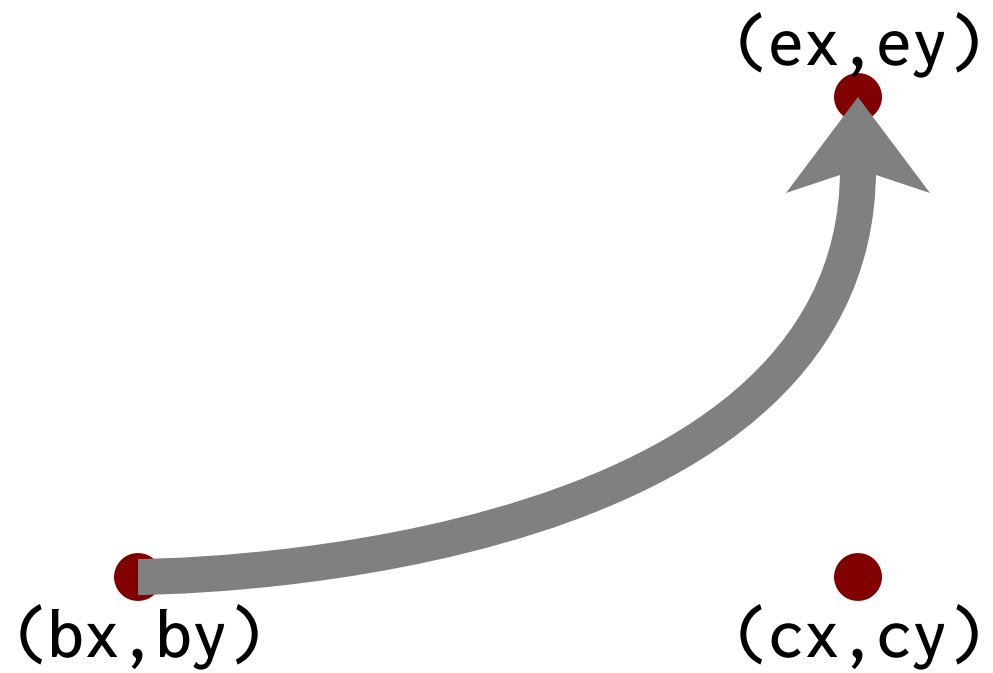
`rcarrow bx by cx cy ex ey lw aw ah color op`



`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"`

$(bx, by)$

$(cx, cy)$

$(ex, ey)$

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



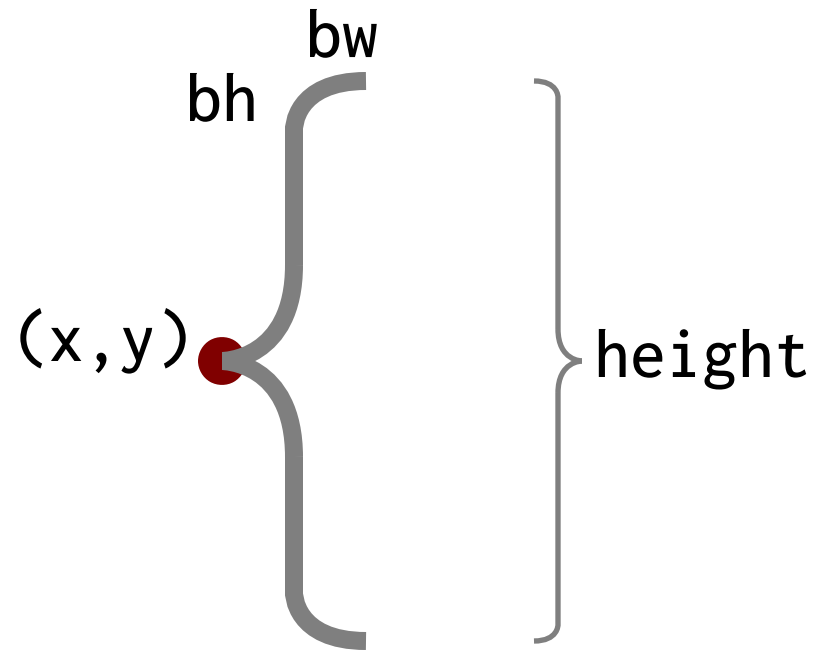
`dcarrow 15 35 30 30 20`



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

# Braces and Brackets

description	keyword	mandatory	optional
Left brace	<b>lbrace</b>	x y height bw bh	lw color op
Right brace	<b>rbrace</b>	x y height bw bh	lw color op
Up brace	<b>ubrace</b>	x y width bw bh	lw color op
Down brace	<b>dbrace</b>	x y width bw bh	lw color op
Left bracket	<b>lbracket</b>	x y width height	lw color op
Right bracket	<b>rbracket</b>	x y width height	lw color op
Up bracket	<b>ubracket</b>	x y width height	lw color op
Down bracket	<b>dbacket</b>	x y width height	lw color op



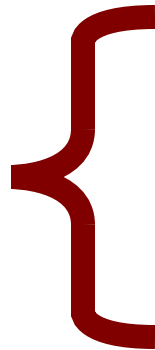
`lbrace` `x` `y` `height` `bw` `bh` `lw` `color` `op`



`lbrace 20 25 20 2 2`

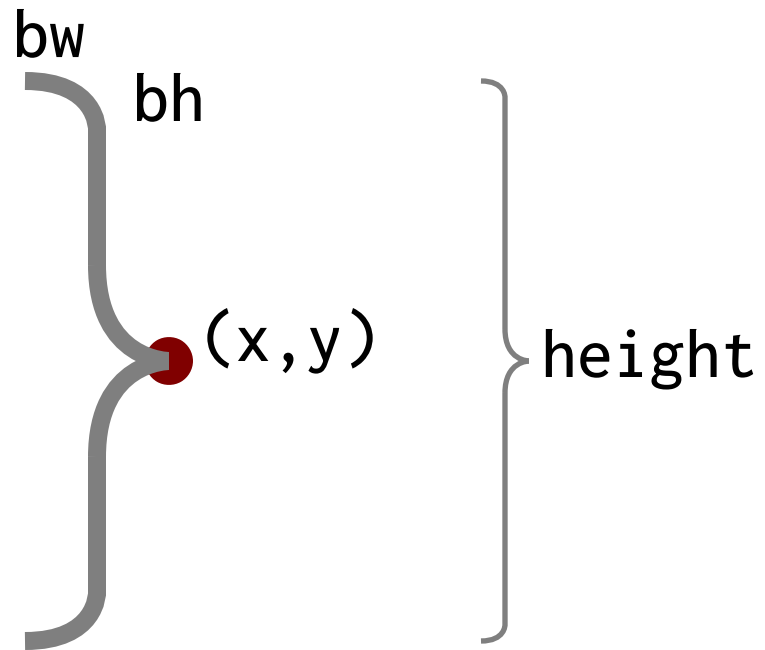


`lbrace 50 25 20 4 4 1`



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





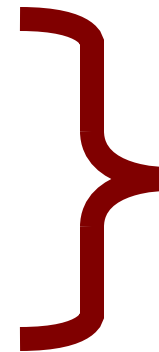
`rbrace` `x` `y` `height` `bw` `bh` `lw` `color` `op`



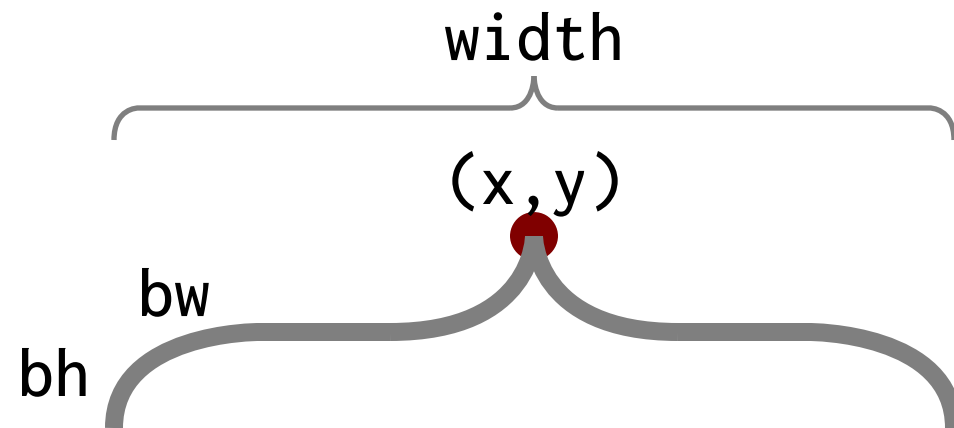
`rbrace 20 25 20 2 2`



`rbrace 50 25 20 4 4 1`



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



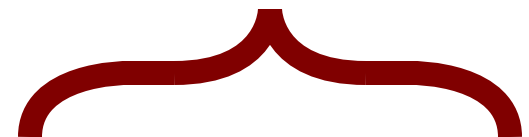
`ubrace`   `x y width bw bh`   `lw color op`



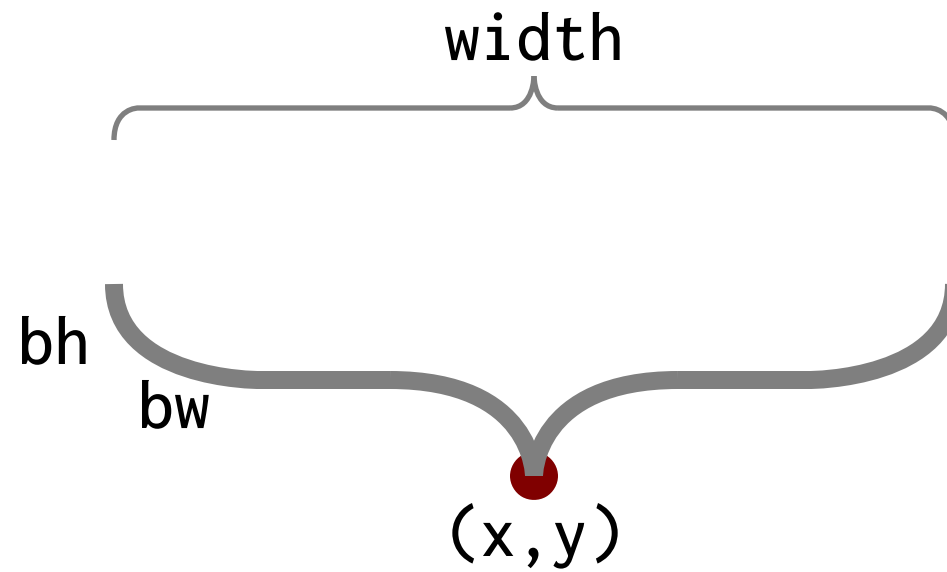
`ubrace 20 25 20 2 4`



`ubrace 50 25 20 4 8 1`



`ubrace 80 25 20 4 8 1 "maroon"`



**dbrace**    **x y width bw bh**    lw color op



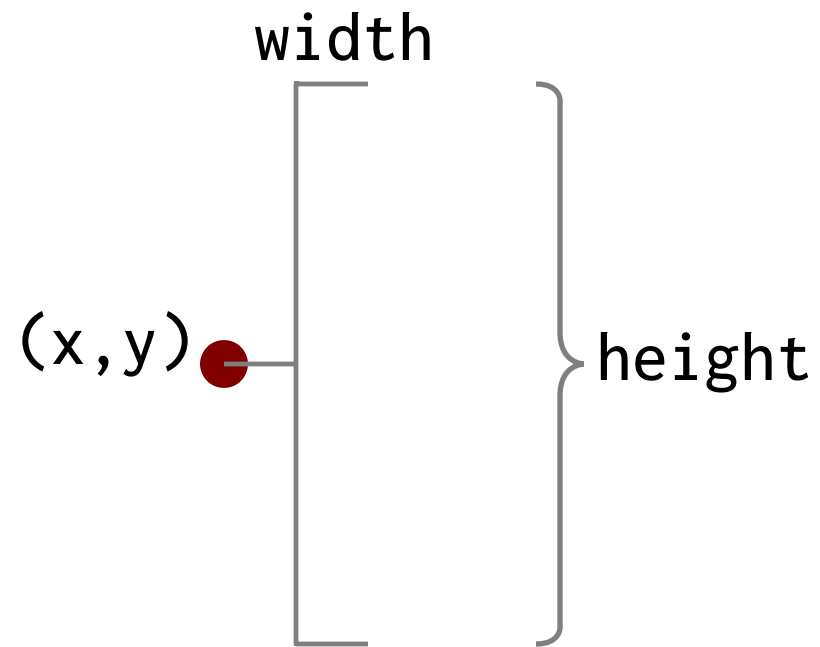
dbrace 20 25 20 2 4



dbrace 50 25 20 4 8 1



dbrace 80 25 20 4 8 1 "maroon"



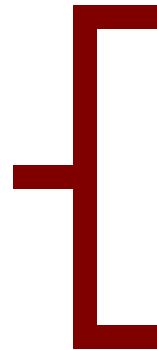
**`lbracket`**   `x`   `y`   `width`   `height`   `lw`   `color`   `op`



`lbrace 20 25 2 20`



`lbracket 50 25 4 20 1`



`lbracket 80 25 6 20 1 "maroon"`

width



(x,y)



height

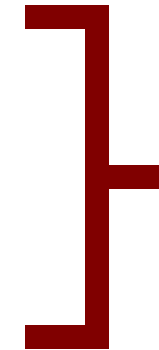
**rbracket** x y width height lw color op



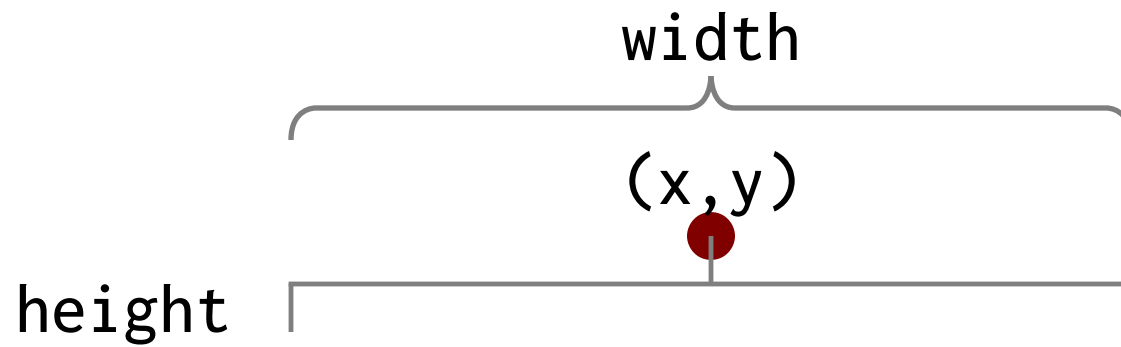
lbrace 20 25 2 20



rbracket 50 25 4 20 1



rbracket 80 25 6 20 1 "maroon"



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



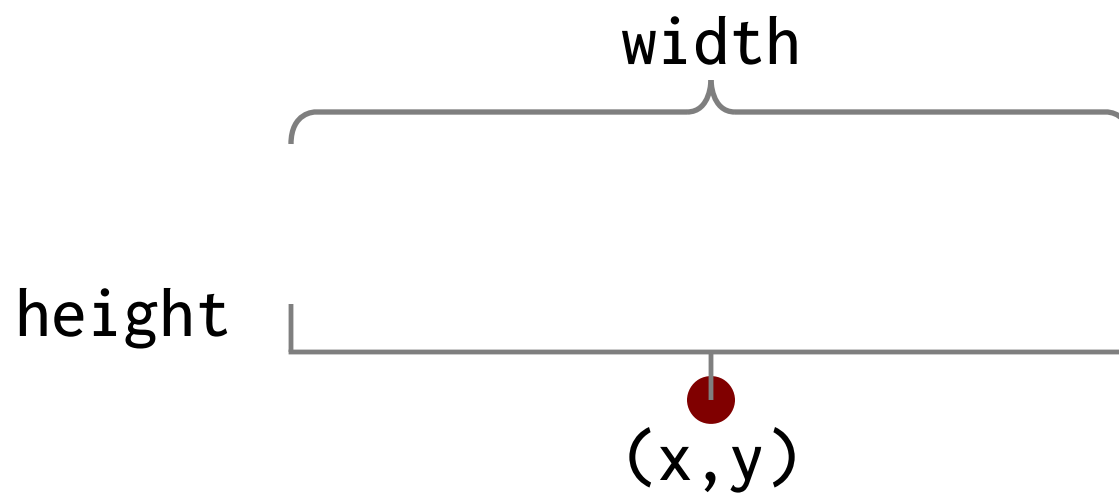
`ubracket 20 25 20 4`



`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, Built-ins, Data, Function and Grid

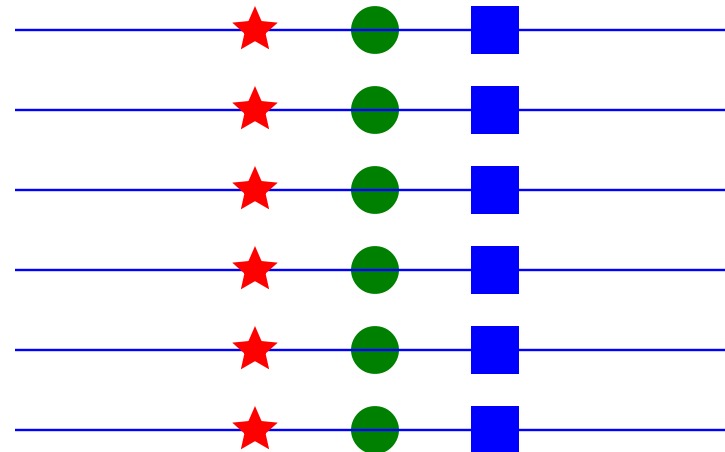
description	keyword	mandatory
Loop	<code>for v=</code>	<code>begin end [increment] ... efor</code>
Coordinate	<code>p=</code>	<code>(x,y)</code>
Polar coordinate (x)	<code>x=polarx</code>	<code>x y radius angle</code>
Polar coordinate (y)	<code>y=polar y</code>	<code>x y radius angle</code>
Polar coordinates	<code>value=polar</code>	<code>x y radius angle</code>
Area	<code>value=area</code>	<code>expression</code>
Formatted text	<code>value=format</code>	<code>fmt expression</code>
Random number	<code>value=random</code>	<code>min max</code>
Square Root	<code>value=sqrt</code>	<code>expression</code>
Value mapping	<code>value=vmap</code>	<code>data min1 max1 min2 max2</code>
Define function	<code>def</code>	<code>name arg1 ... argn ... edef</code>
Call function	<code>func</code>	<code>"file" arg1 ... argn</code>
In-line data	<code>data</code>	<code>"file" ... edata</code>
Objects on a grid	<code>grid</code>	<code>"file" x y hspace vspace edge</code>

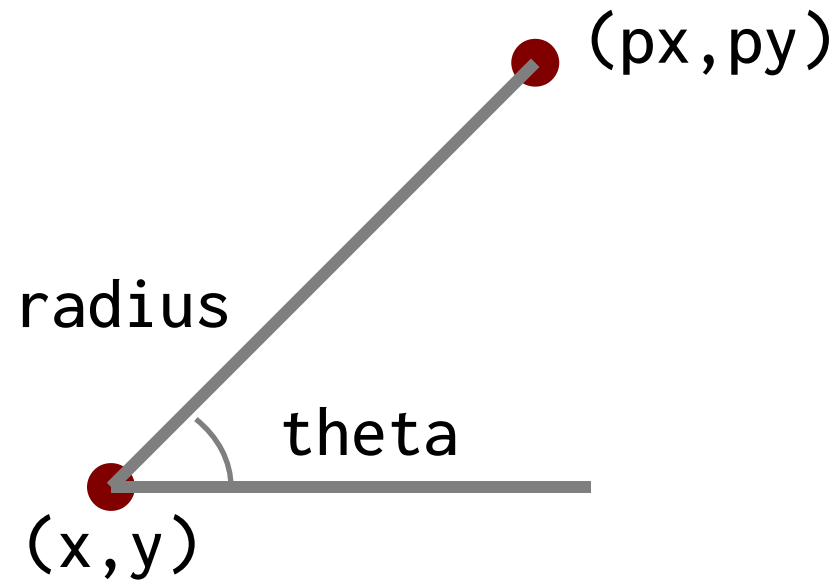


```
for v=begin end [increment]  
...items to repeat using v  
efor
```

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

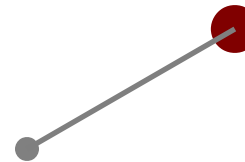
```
for v=10 35 5  
  hline 50 v 30 0.1 "blue"  
  star 60 v 5 1 0.4 "red"  
  circle 65 v 2 "green"  
  square 70 v 2 "blue"  
efor
```

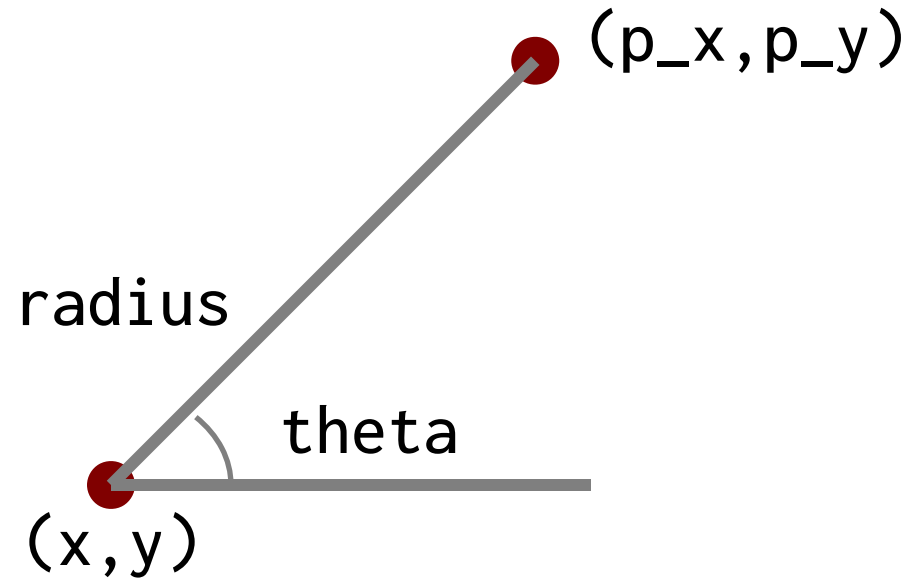




```
px=polarx x y radius theta  
py=polary x y radius theta
```

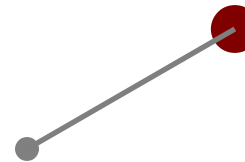
```
cpx=60  
cpy=20  
px1=polarx cpx cpy 10 30  
py1=polary cpx cpy 10 30  
line cpx cpy px1 py1  
circle cpx cpy 1 "gray"  
circle px1 py1 2 "maroon"
```





`p=polar x y radius theta`

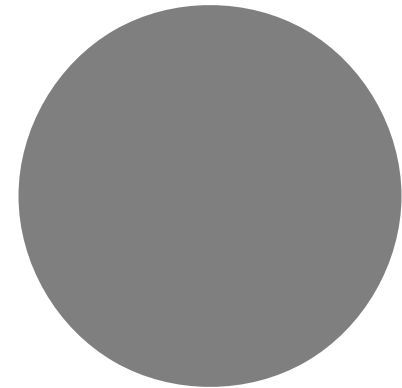
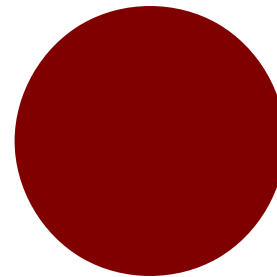
```
cpx=60  
cpy=20  
point=polar cpx cpy 10 30  
line cpx cpy point_x point_y  
circle cpx cpy 1 "gray"  
circle point_x point_y 2 "maroon"
```



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

v1=100.3

v2=200.234

title=format "%.2f Million (USD)" v1

subtitle=format "Total value: %.2f" v1+v2

ctext title      80 30 4 "sans" "maroon"


ctext subtitle 80 20 3 "sans" "gray"

100.30 Million (USD)


Total value: 300.53




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"
```

a=10

b=71

x=sqrt 4

← x = 2

y=sqrt a

← y = 3.1622776

n=sqrt a+b

← n = 9

value=**sqrt** number or expression

```
a_squared=10*10
```

```
b_squared=20*20
```

```
c=sqrt a_squared + b_squared
```

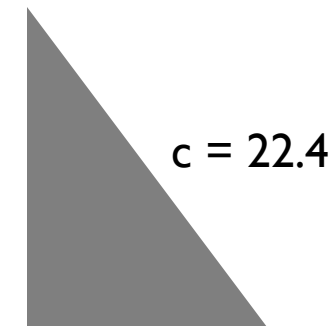
```
w=70+10
```

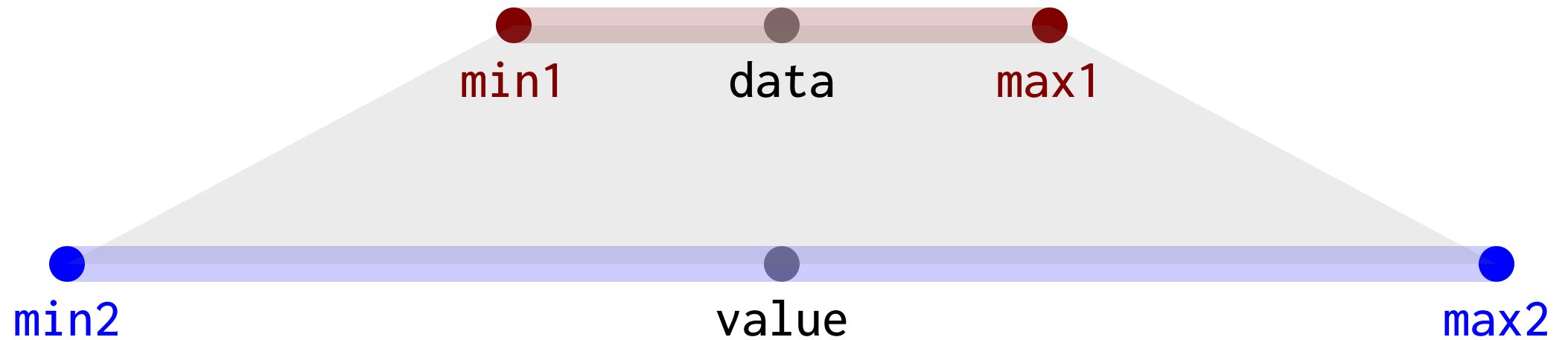
```
h=15+20
```

```
polygon "70 w 70" "15 15 h"
```

```
clabel=format "c = %.1f" c
```

```
text clabel 76 25 2
```





`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"
```





```
func "doit.dsh" 50 20 2.5 "hello"
```

definition in doit.dsh

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



result of function call

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

```
func "file" arg1 ... argn
```

calling the function  
call again

```
func "foo.dsh" 50 30 5 "calling the function"  
func "foo.dsh" 50 20 4 "call again"
```

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

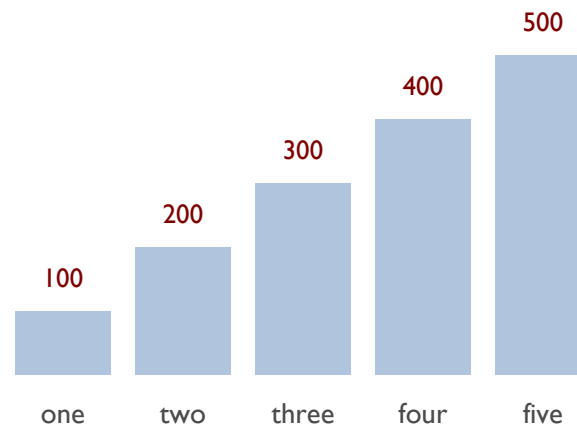
} data values

```
data "file" ... edata
```

```
data "test.d"
  one 100
  two 200
  three 300
  four 400
  five 500
```

```
edata
```

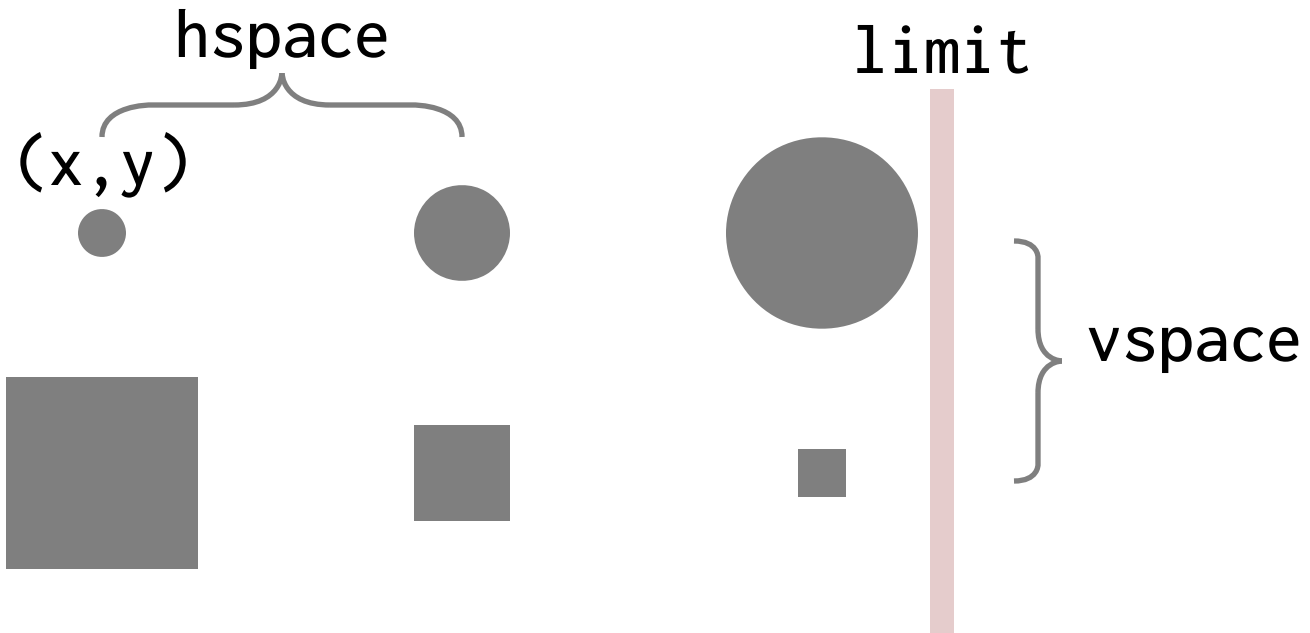
```
dchart -bar -left 50 -bottom 15 -right 70 -top 35 "test.d"
```



file

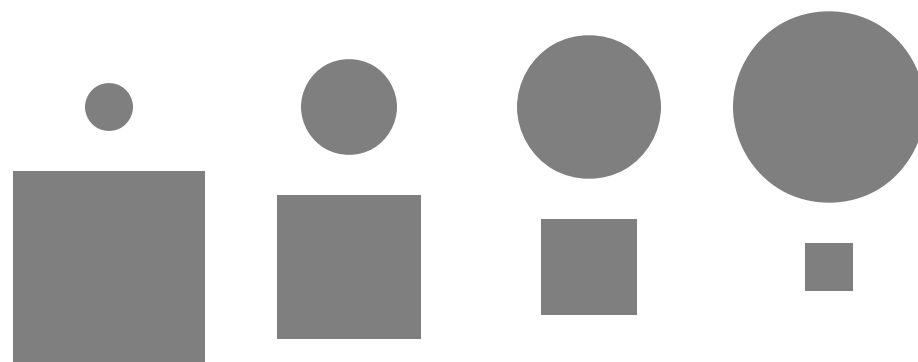


```
circle x y 2
circle x y 4
circle x y 8
square x y 8
square x y 4
square x y 2
```



**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

# Charts

description	keyword	arguments
Charts	<b>dchart</b>	options "file" (see next page)
Chart Legends	<b>legend</b>	"text" x y size font color

## Chart Types

-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

## Chart Elements

-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

## Position and Scaling

-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

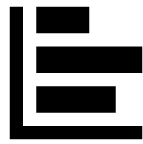
## Measures and Attributes

-bgcolor	white	background color
-barwidth	computed from data size	barwidth
-color	lightsteelblue	data color
-csvcol	label1,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 %

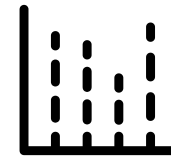
# dchart types



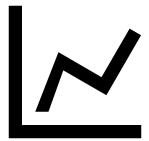
Column



Bar



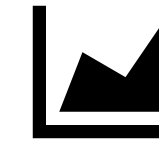
Dot



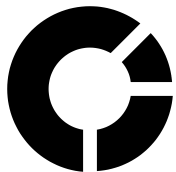
Line



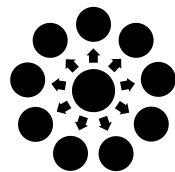
Scatter



Area



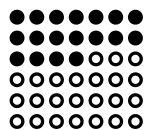
Donut/Pie



Radial



Pmap



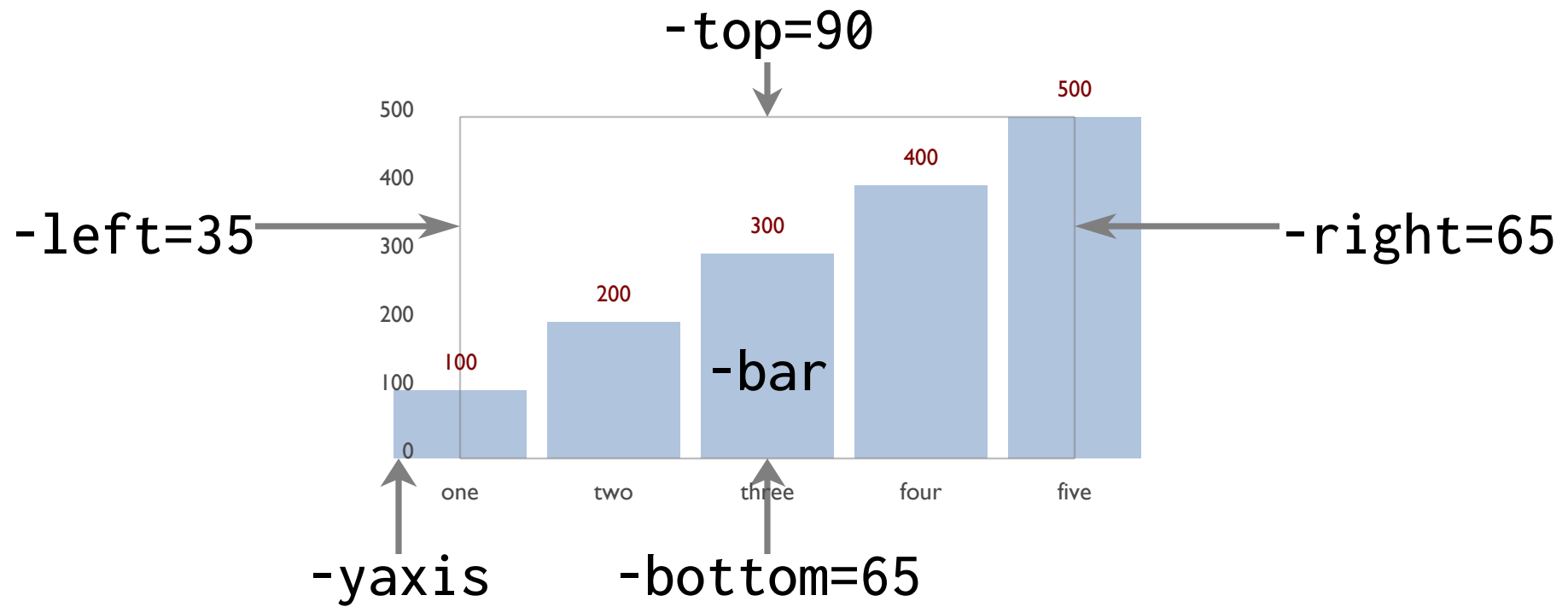
Waffle/Lego



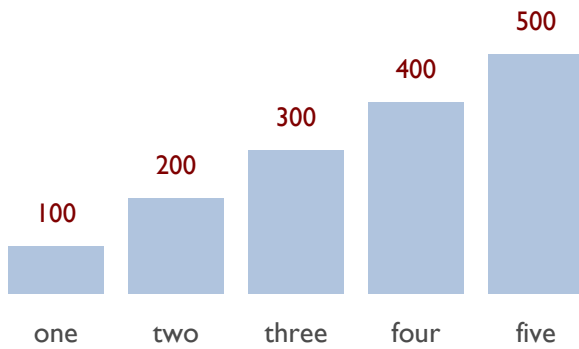
Fan



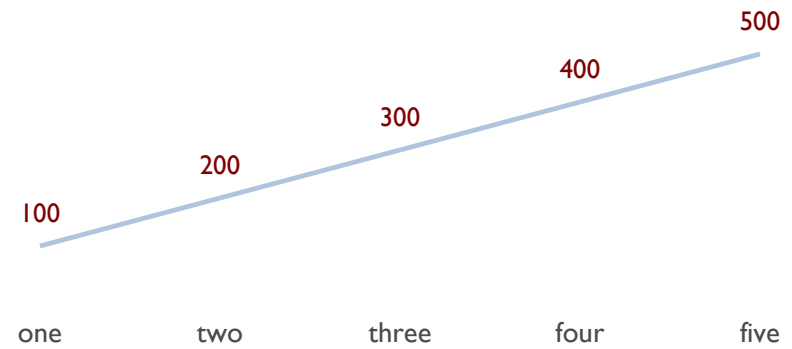
Bowtie



## 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"
```



■ My text  
(x, y)

legend 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"
```