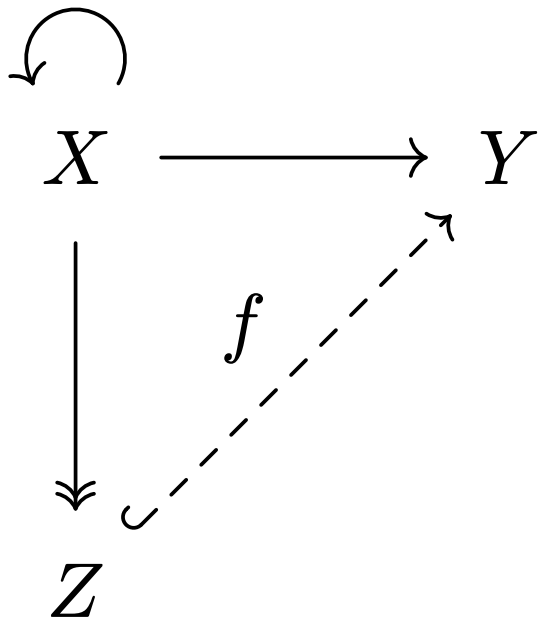
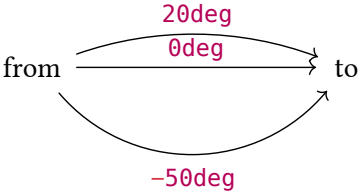
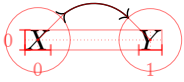
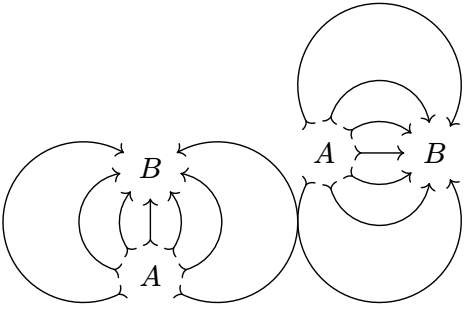
Connectors

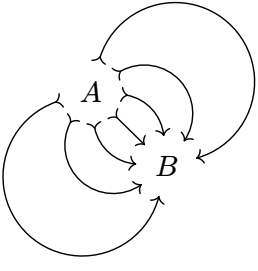


Arc connectors





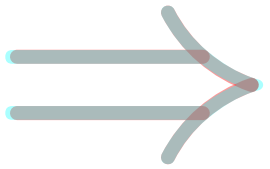


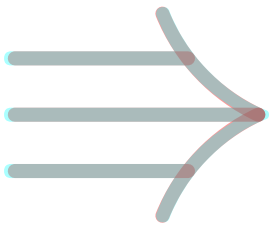


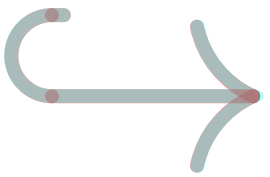
Matching math arrows

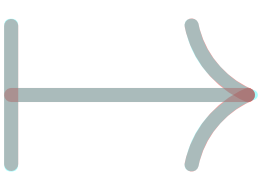
Compare to $\rightarrow$, $\Rightarrow$ $\Rrightarrow$ $\twoheadrightarrow$, $\hookrightarrow$, $\mapsto$.

Red is our output; cyan is reference symbol in default math font.









Double and triple lines

Diagram $A \xrightarrow{f} B$ and equation $A \to B$.

Diagram $A \xRightarrow{f} B$ and equation $A \Rightarrow B$.

Diagram $A \xRightarrow{f} B$ and equation $A \Rrightarrow B$.

Arrow head shorthands

"->" =

"<-" =

">-<" =

"<->" =

"<=>" =

"<==>" =

"|->" =

"|=>" =

">->" =

"<<->>" =

">>-<<" =

">>>-stealth" =

"hook->" =

"hook'--hook" =

"|=|" =

"||-||" =

"|||-|||" =

"/--\\" =

"\\=\\" =

"/=/" =

"x-X" =

">>-<<" =

"harpoon-harpoon'" =

"harpoon'-<<" =

"<--hook'" =

"|..|" =

"hooks--hooks" =

"o-O" =

"O-o" =

"*-@" =

"o==O" =

"||->>" =

"<|-|>" =

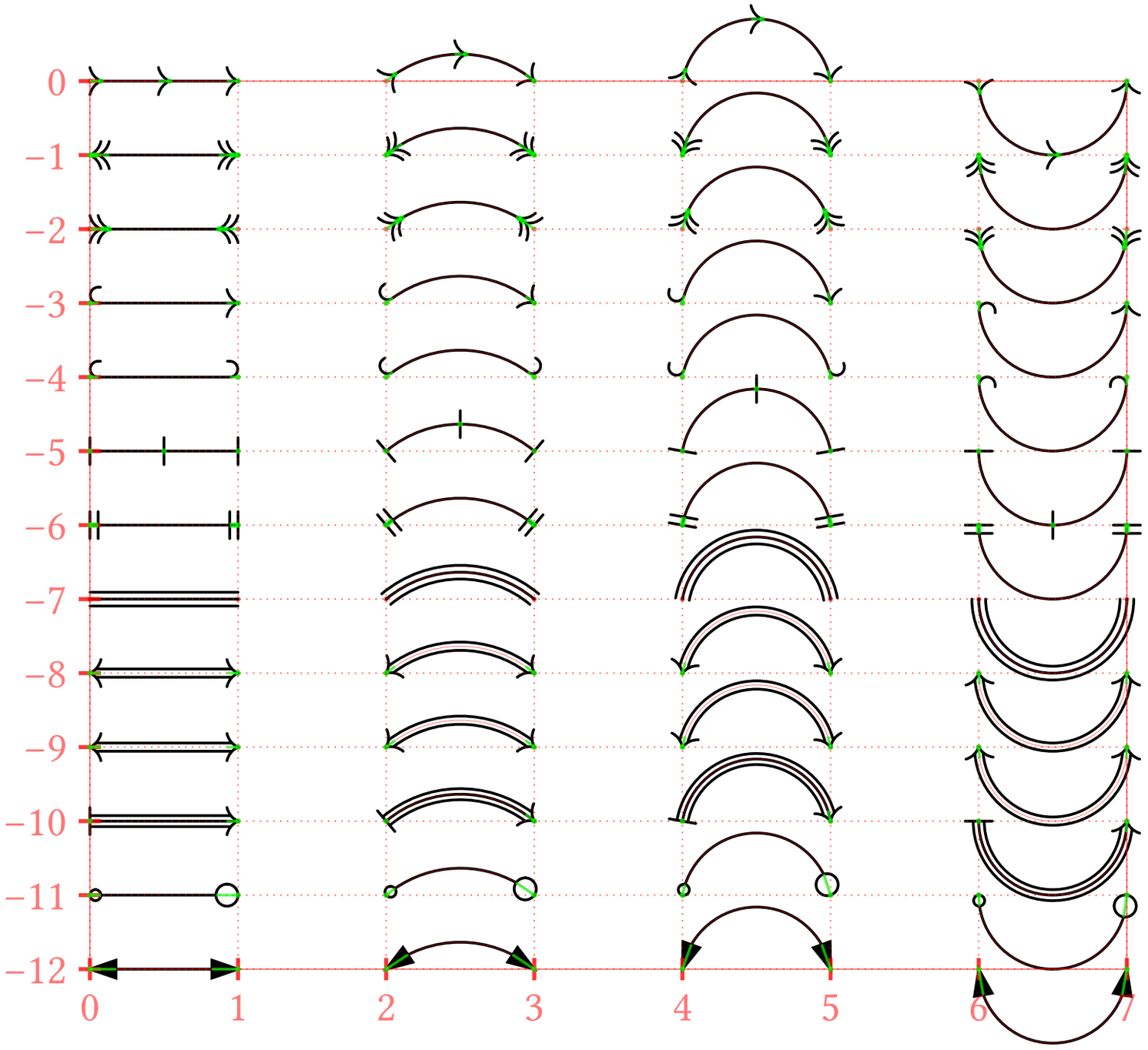
"|>-<|" =

"-|-" =

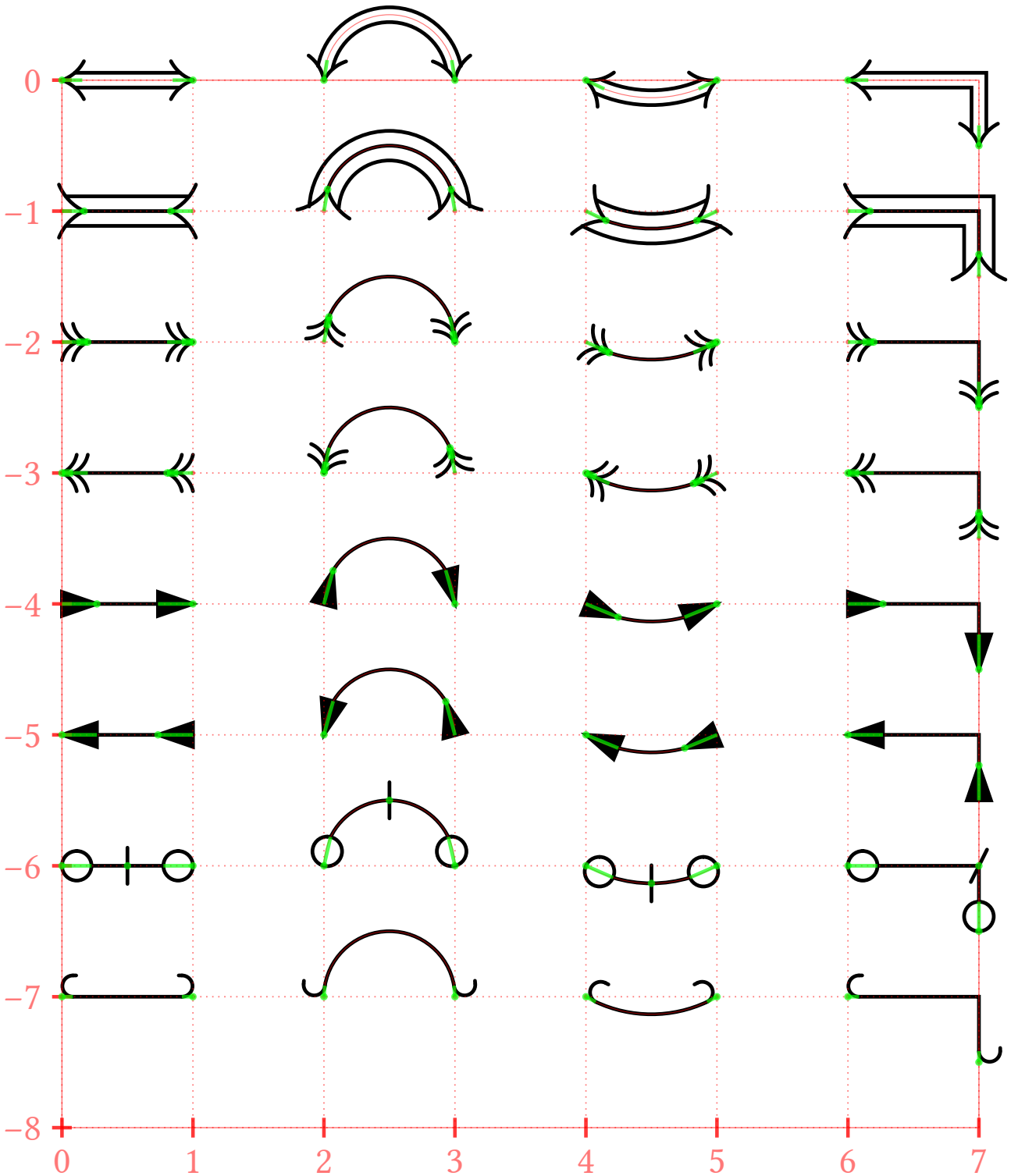
"hook-/->" =

"stealth-stealth" =

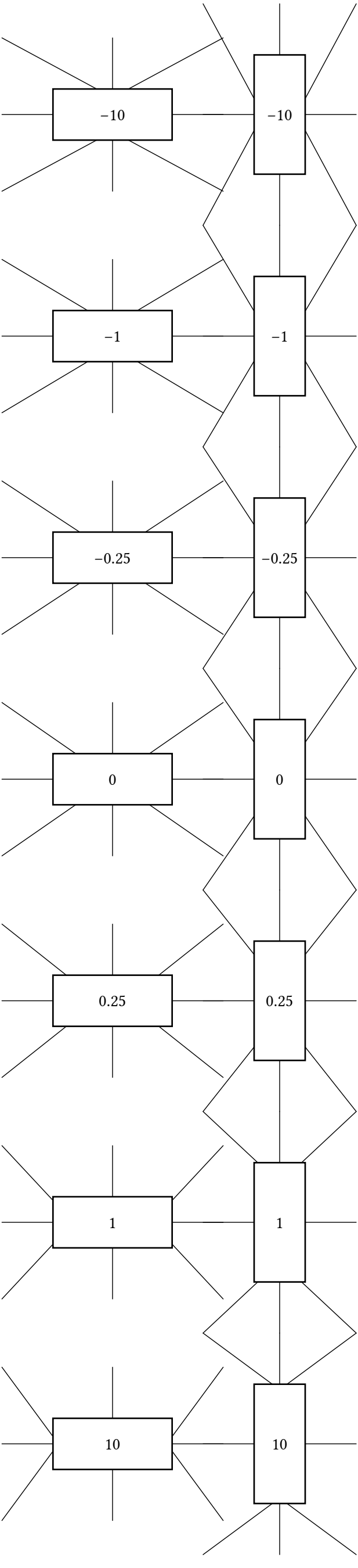
Bending arrows



Fine mark angle corrections

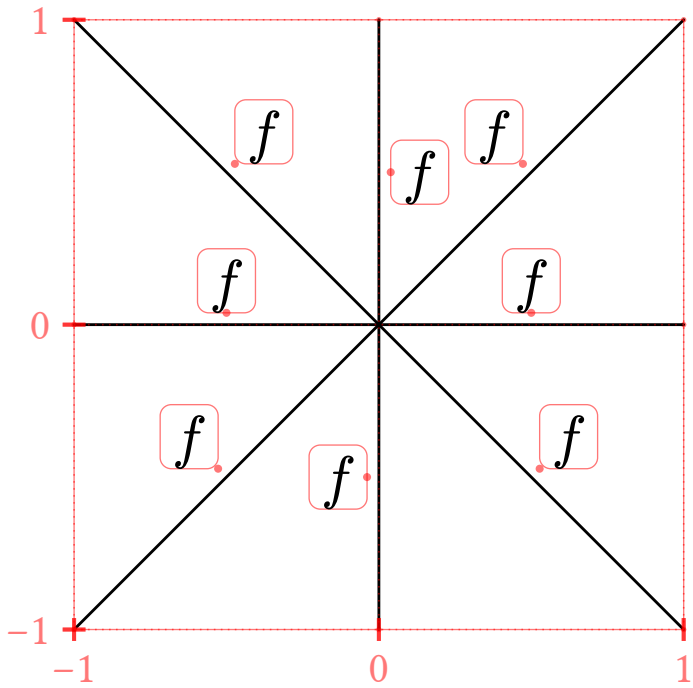


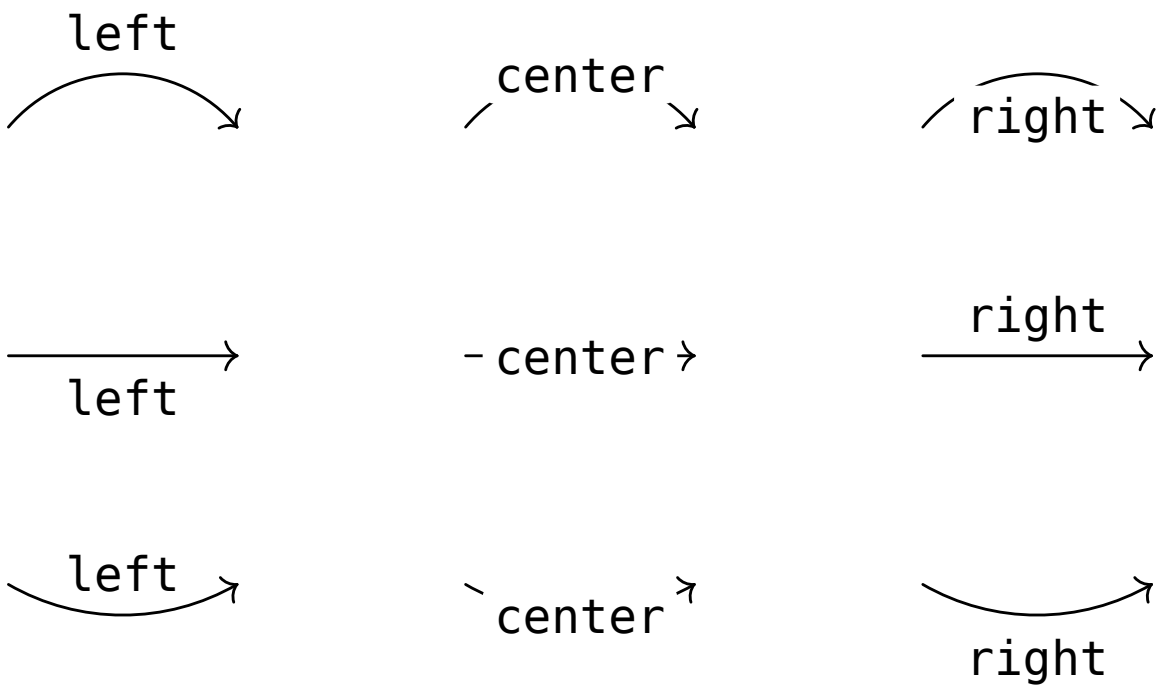
Defocus adjustment



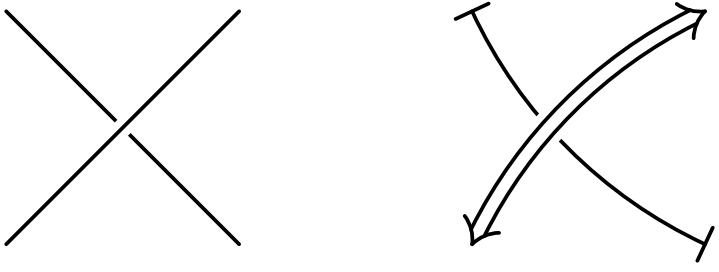
Label placement

Default placement above the line.





Crossing connectors



`edge()` argument shorthands

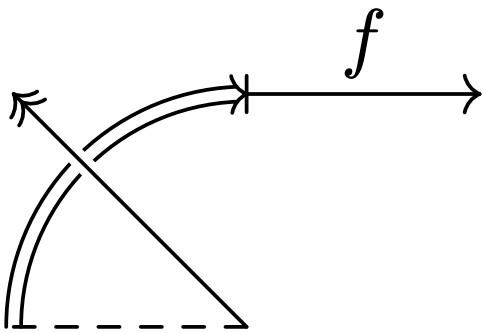
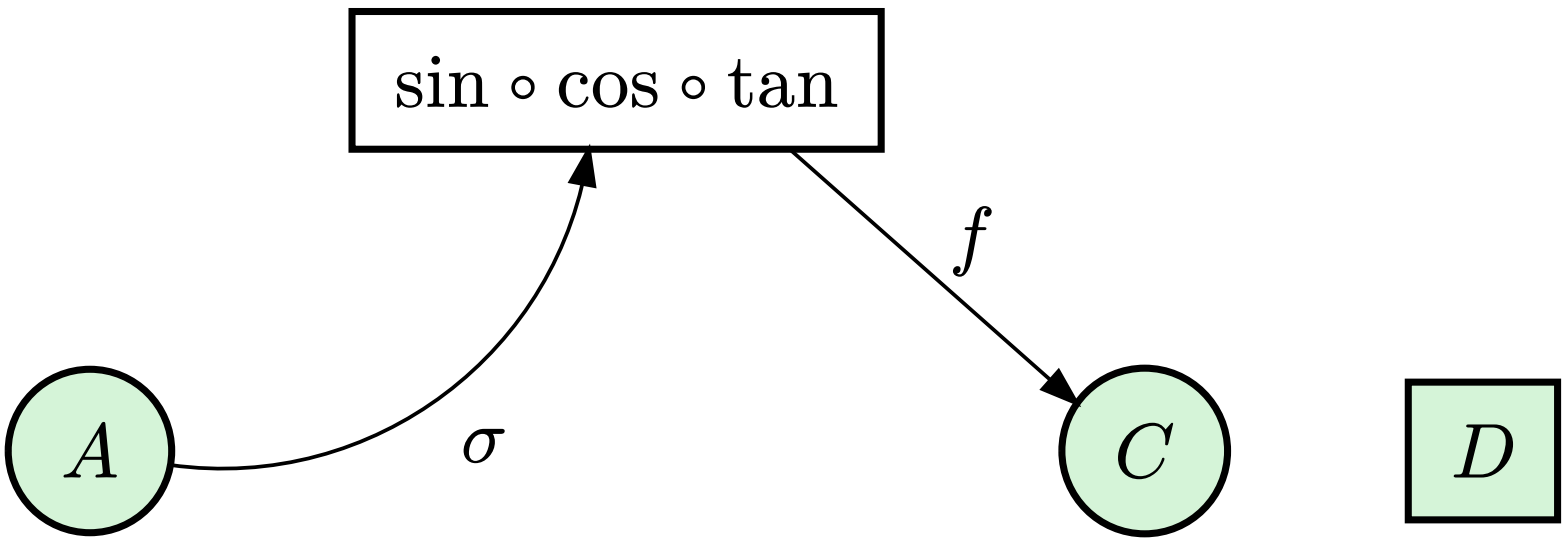
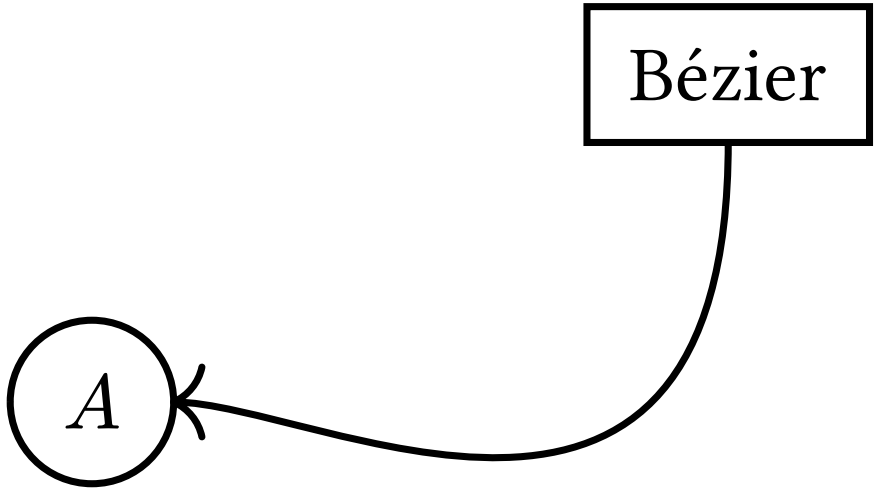


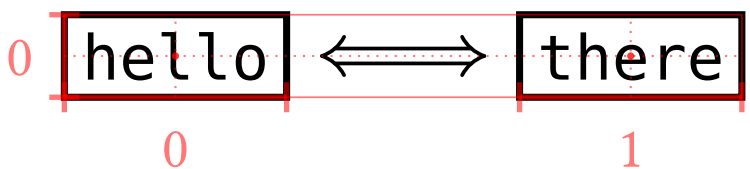
Diagram-level options



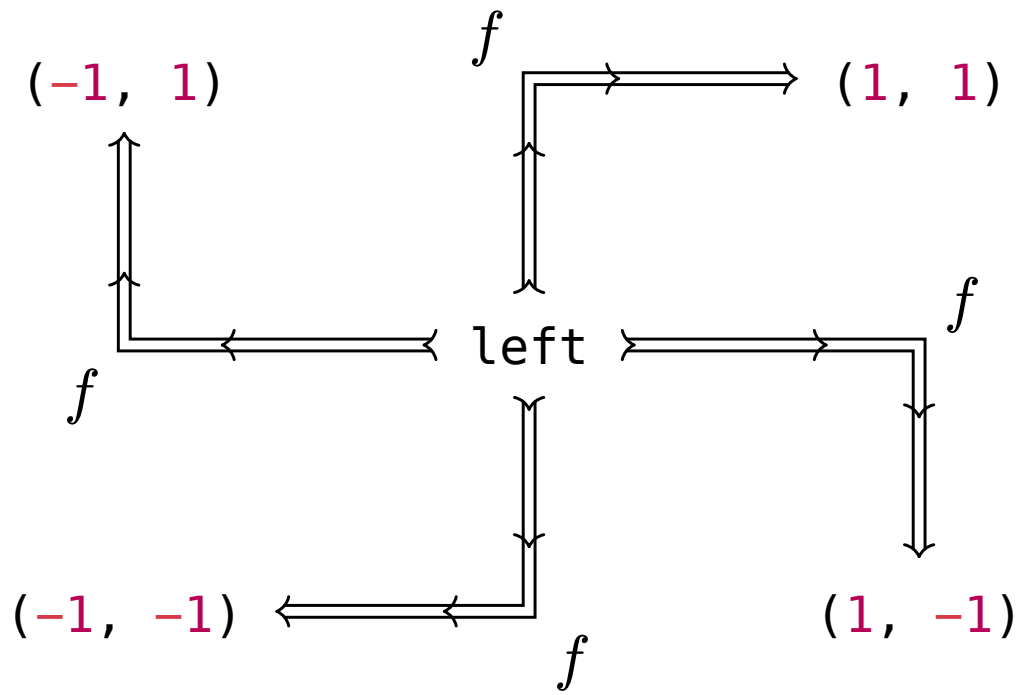
CeTZ integration

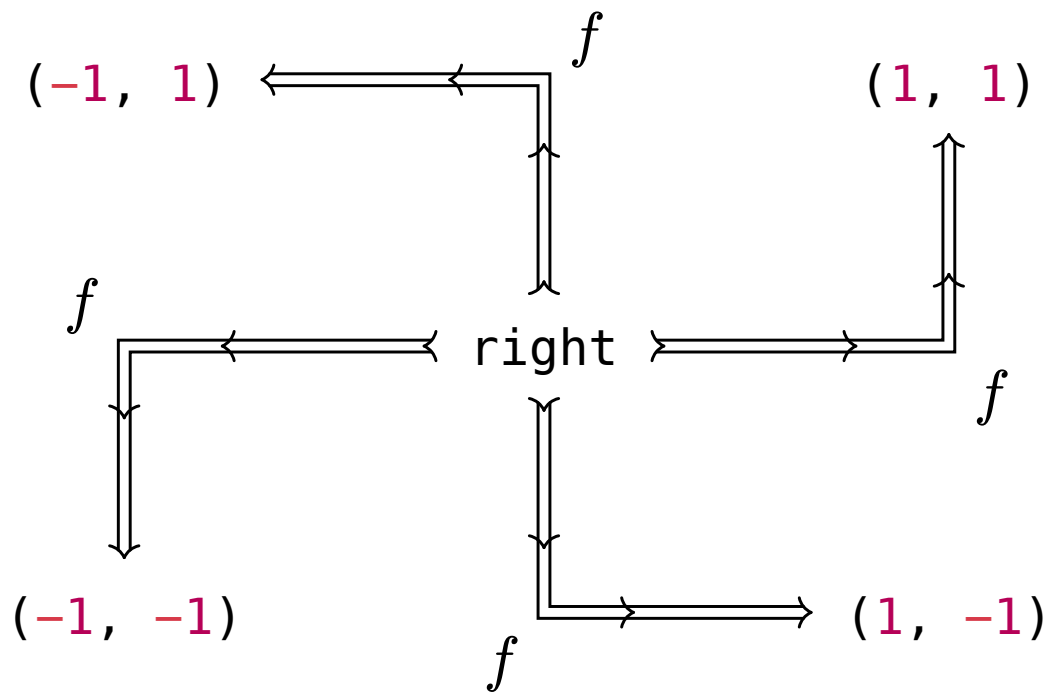


Node bounds

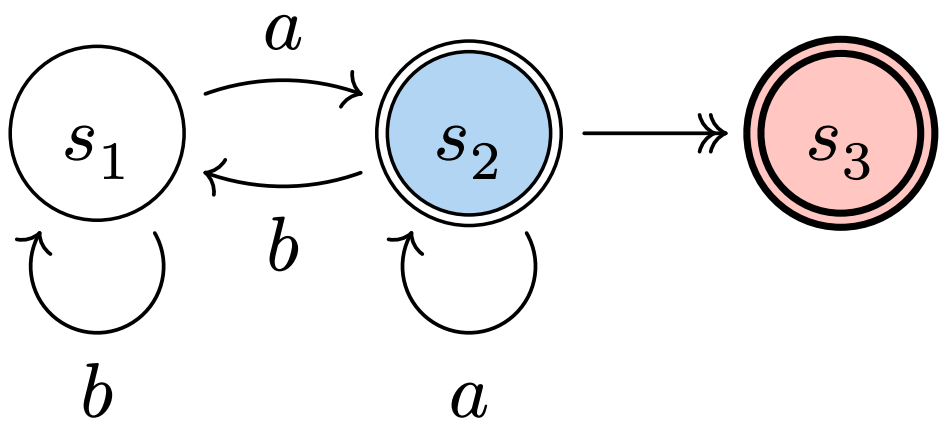


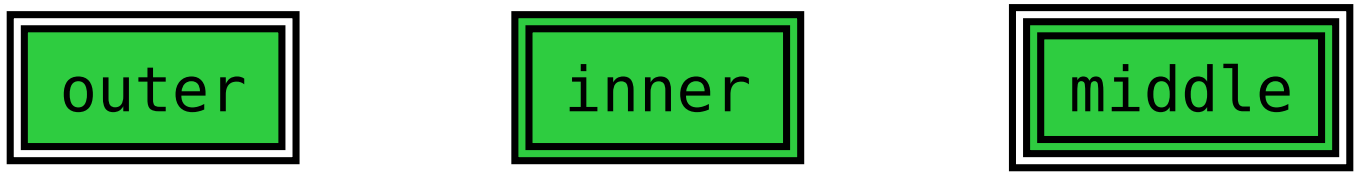
Corner edges



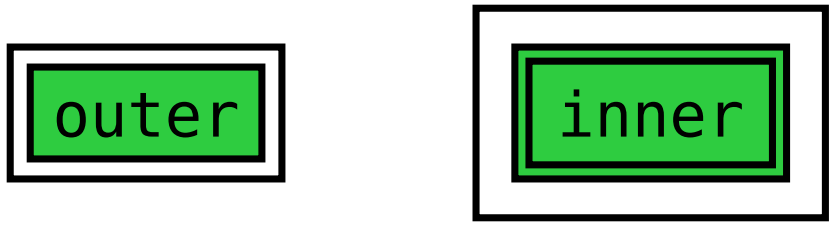


Double node strokes



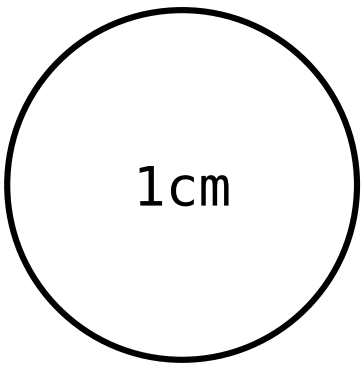


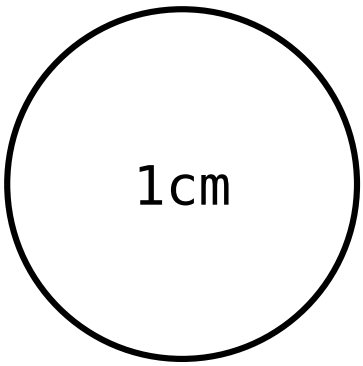
Relative and absolute extrusion lengths

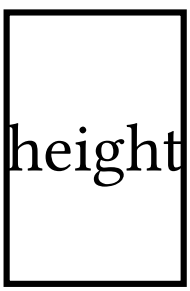


Custom node sizes

Make sure provided dimensions are exact, not affected by node `inset`.



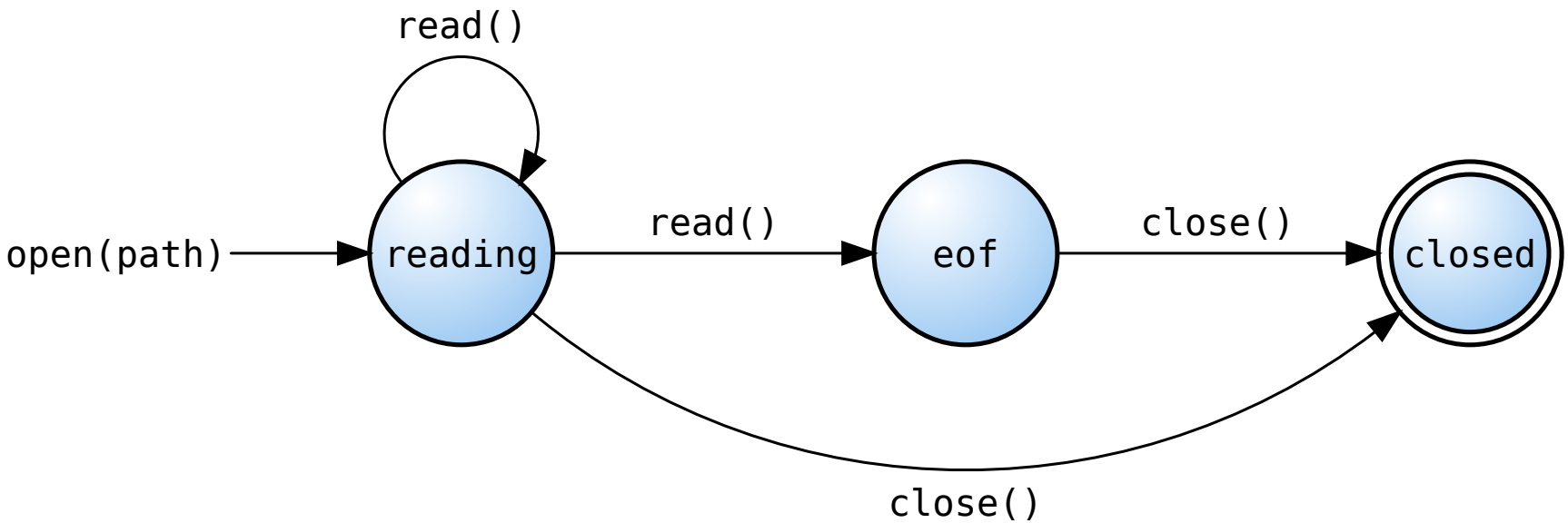




Example

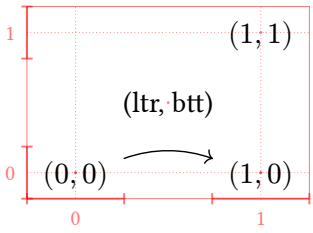
Make sure node or edge labels don't pick up equation numbers!

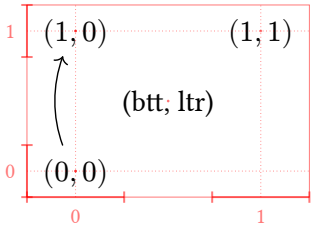
$$a^2 \tag{1}$$

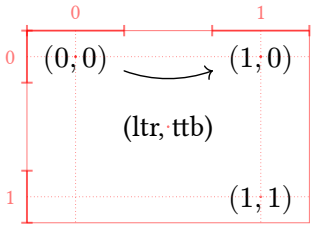


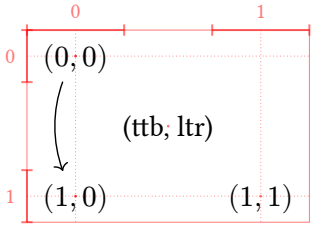
$$b^2 \tag{2}$$

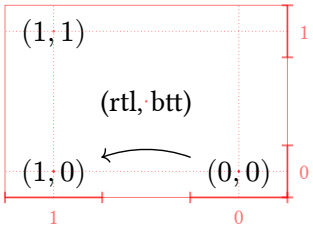
Funky axes

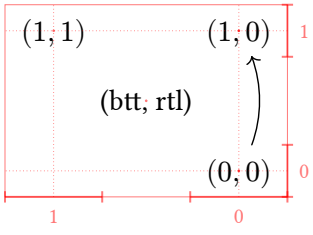

1
(1; 1)
(ltr, btt)
0
(0; 0)
(1; 0)
0
1

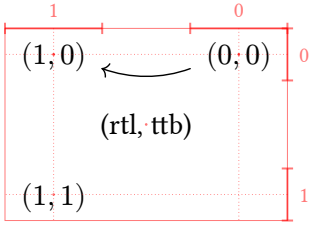

1
(1; 0)
(1; 1)
(btt, ltr)
0
(0; 0)
0
1

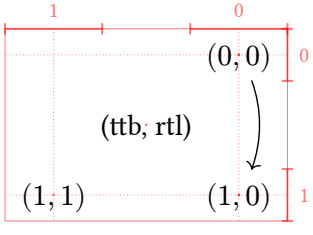

0
1
0
(0; 0)
(1; 0)
(ltr, ttb)
1
(1; 1)


0
1
0
(0; 0)
(ttb, ltr)
1
(1; 0)
(1; 1)


(1; 1)
1
(rtl, btt)
(1; 0)
(0; 0)
0
1
0


(1; 1)
(1; 0)
1
(btt, rtl)
(0; 0)
0
1
0


1
0
(1; 0)
(0; 0)
0
(rtl, ttb)
(1; 1)
1


1
0
(0; 0)
0
(ttb, rtl)
(1; 1)
(1; 0)
1

?

```
(size: 2, fill: true, outer-len: 4,
kind: "circle")
```



```
(
  [G],
  [ ],
  metadata(value: (kind: "edge",
options: ("r", "→", [f]))),
  [ ],
  metadata(value: (kind: "edge",
options: ("d", "↠", "π"))),
  [ ],
  align-point(),
  [ ],
  [(
    op(text: [im], limits: false),
    lr(body: [([(], [f], [)])]),
  )],
  [ ],
  linebreak(),
  [ ],
  [G],
  [ ],
  [/],
  [ ],
  [(
    op(text: [ker], limits: false),
    lr(body: [([(], [f], [)])]),
  )],
  [ ],
  metadata(
    value: (
      kind: "edge",
      options: ("ur", "→", accent(base:
[f], accent: "\u{303}")),
    ),
  ),
)
```