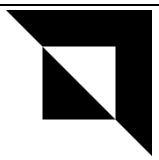
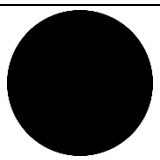
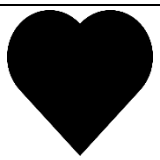
Runes Library

- Use `show(rune)` to display the rune.

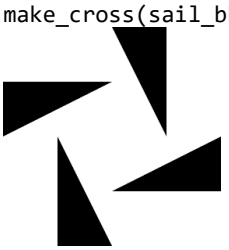
Rune	Reference Image
`black_bb`	
`blank_bb`	
`rcross_bb`	
`sail_bb`	
`corner_bb`	
`nova_bb`	
`circle_bb`	
`heart_bb`	
`pentagram_bb`	
`ribbon_bb`	

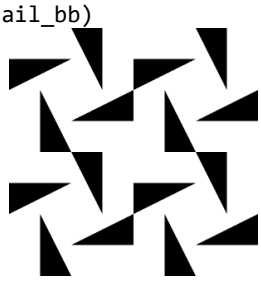
Transformation Functions

- `scale(ratio, shape)`: Scales the `shape` to the `ratio`
- `scale_independent(ratio_x, ratio_y, shape)`: Scales the `shape` by `ratio_x` along the x-axis and `ratio_y` along the y-axis
- `translate(x, y, shape)`: Shifts the `shape` by `x` amount along the x-axis and `y` amount along the y-axis
- `rotate(rad, shape)`: Rotates the `shape` by `rad` amount in radians in the *counter-clockwise direction*
- `quarter_turn_right(shape)`: Rotates the `shape` clockwise by $\frac{\pi}{2}$
- `quarter_turn_left(shape)`: Rotates the `shape` counter-clockwise by $\frac{\pi}{2}$
- `turn_upside_down(shape)`: Rotates the `shape` by π
- `flip_vert(shape)`: Flips the `shape` vertically along the x-axis
- `flip_horiz(shape)`: Flips the `shape` horizontally along the y-axis
- `stack(shape1, shape2)`: Stacks `shape1` onto `shape2` with equal proportions (i.e. $\frac{1}{2}$)
- `stack_frac(frac, shape1, shape2)`: Stacks `shape1` onto `shape2` such that `shape1` takes up `frac` amount of space in the rune
- `stackn(n, shape)`: Stacks `n` times of `shape` with equal proportions (i.e. $\frac{1}{n}$)
- `beside(shape1, shape2)`: Puts `shape1` to the left of `shape2` with equal proportions
- `beside_frac(frac, shape1, shape2)`: Puts `shape1` to the left of `shape2` such that `shape1` takes up `frac` amount of space in the rune
- `make_cross(shape)`:
  Example
  `sail_bb`

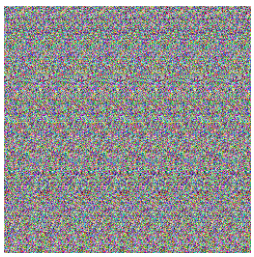
  `make_cross(sail_bb)`

  
- `repeat_pattern(n, pattern, shape)`: Repeats the transformation function `pattern` applied to the `shape` `n` times
  Example
  `repeat_pattern(2, make_cross, sail_bb)`

  
- `overlay(shape1, shape2)`: Overlays `shape1` on top of `shape2` with equal proportions in the z-axis `[3D rune]`
- `overlay_frac(frac, shape1, shape2)`: Overlays `shape1` on top of `shape2` such that `shape1` takes up `frac` amount of space in the rune in the z-axis `[3D rune]`

3D Display Functions

- `stereogram(shape)`:
  Example
  `stereogram(heart_bb)`



- `anaglyph(shape)`:
  Example
  `anaglyph(heart_bb)`

- `hollusion(shape)`

Color Functions

- `color(shape, r, g, b)`: Colors the `shape` with RGB color. Values range from 0 to 255.
- `color(shape, r, g, b, a)`: Same as above, but with alpha value to specify opaqueness. Values of `a` range from 0 (transparent) to 1 (opaque).
- ***`color`***`(shape)`: Colors the `shape` with the standard color function
- `random_color(shape)`: Colors the `shape` with a random color

Color	r	g	b	a
red	244	67	54	1
pink	233	30	99	1
purple	170	0	255	1
indigo	63	81	181	1
blue	33	150	243	1
green	76	175	80	1
yellow	255	235	59	1
orange	255	152	0	1
brown	121	85	72	1
black	0	0	0	1
white	255	255	255	1

Standard color function
red(shape)
pink(shape)
purple(shape)
indigo(shape)
blue(shape)
green(shape)
yellow(shape)
orange(shape)
brown(shape)
black(shape)
white(shape)