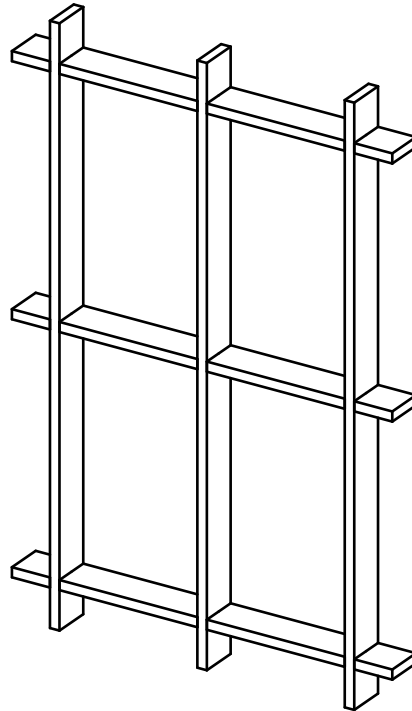


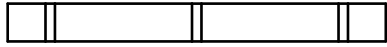
Shoji frame



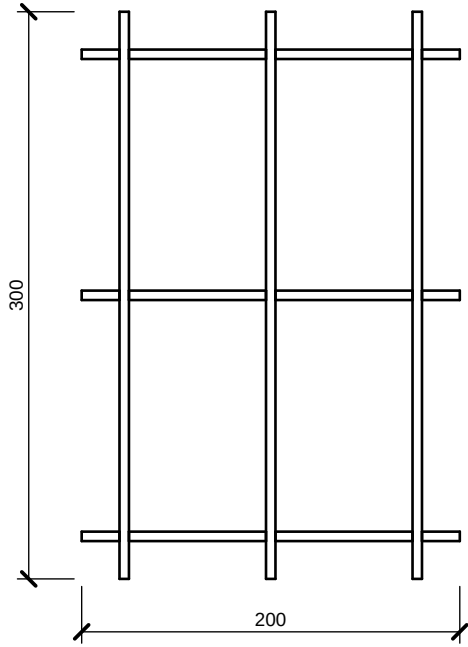
parts	6 (2 different)
overall	200 x 20 x 300 (W x D x H)
material	oak
date	2026-10-25
made with	woodwright 0.9.0

Contents

Drawings	3
Assembly	3
Parts	4
1 vertical	5
2 horizontal	6
Cut list	7
Appendix: source	8



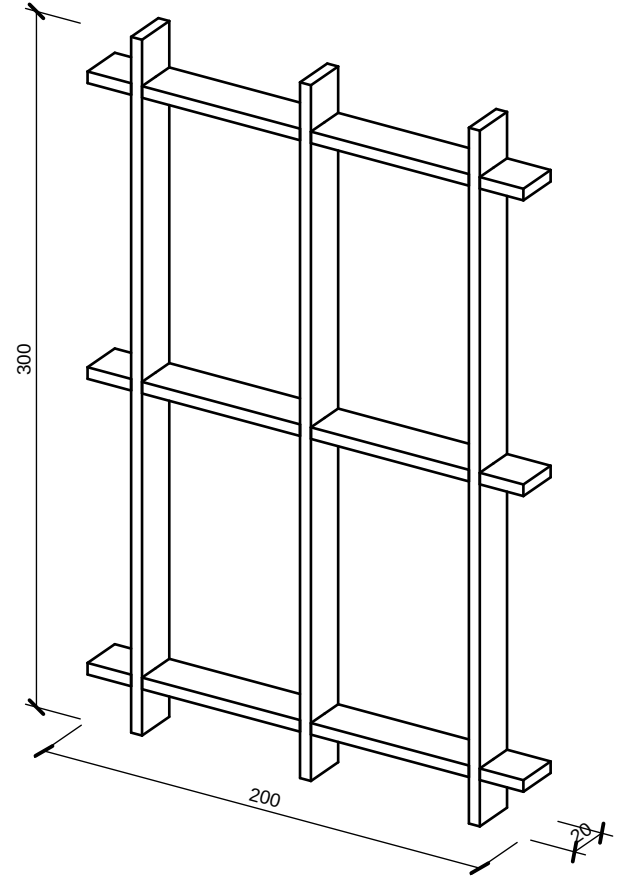
TOP



FRONT

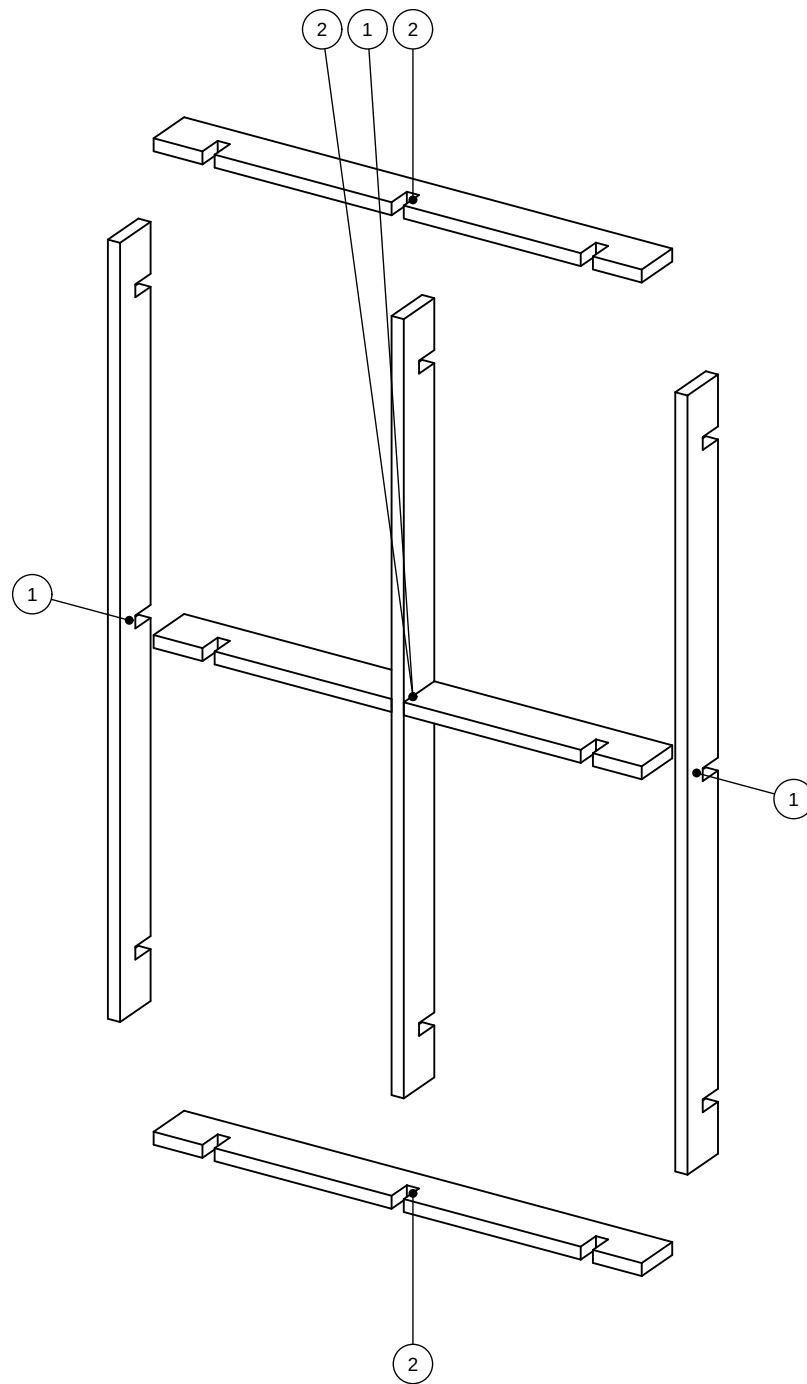


RIGHT SIDE



ISOMETRIC

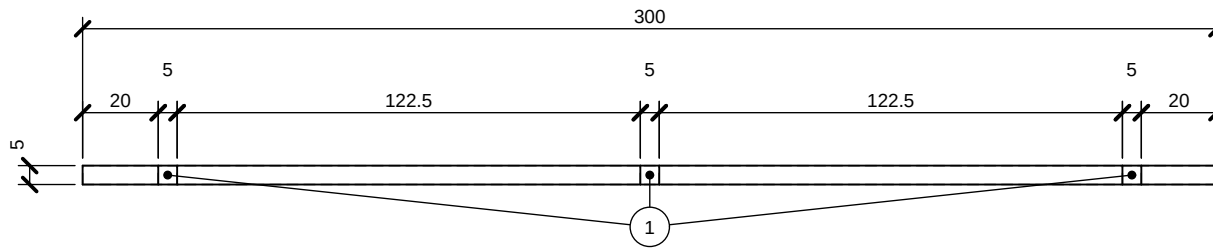
assembly	shoji_frame
parts	6 (2 different)
overall	200 x 20 x 300 (W x D x H)
material	oak
scale	1:4, mm



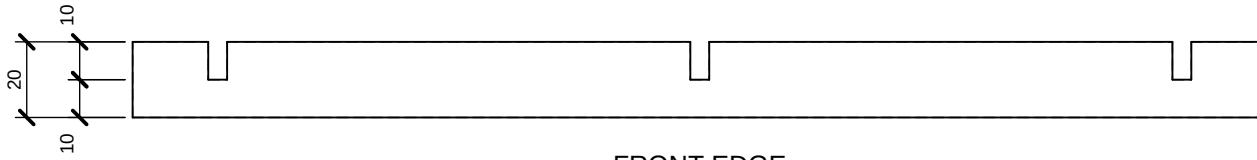
EXPLODED

no.	part	qty	cut size (L x W x T)	material
1	vertical	3	300 x 5 x 20	oak
2	horizontal	3	200 x 5 x 20	oak

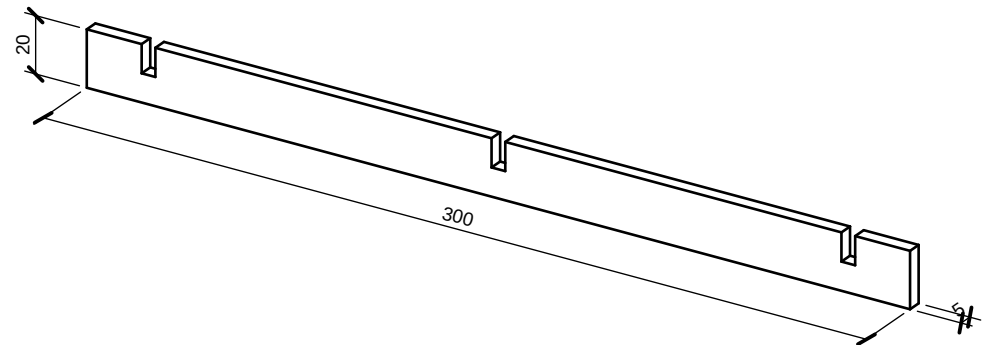
assembly	shoji_frame
parts	6 (2 different)
overall	200 x 20 x 300 (W x D x H)
material	oak
scale	not to scale, mm



TOP FACE



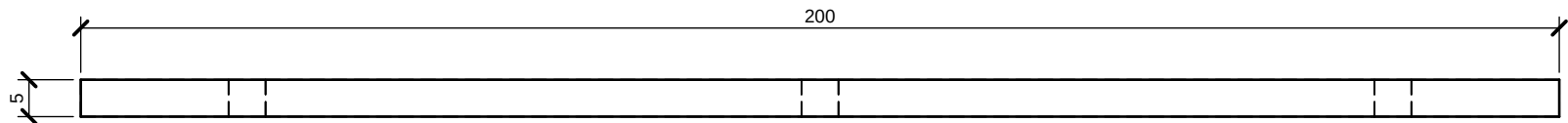
FRONT EDGE



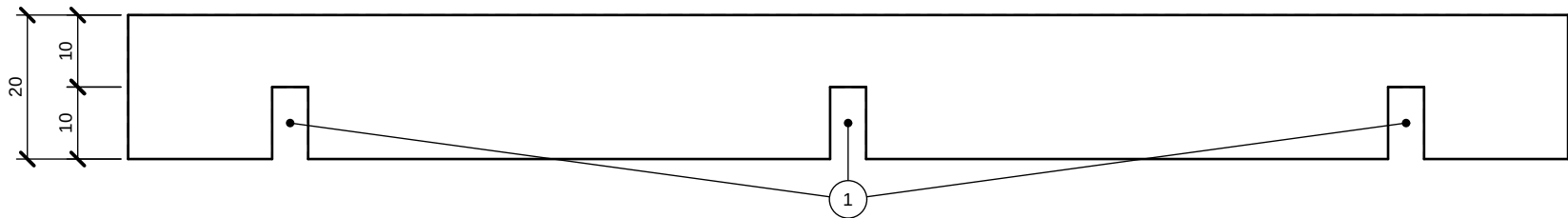
ISOMETRIC

3 pieces: vertical_1, vertical_2, vertical_3
 1 3 x dado 5 mm, 10 deep, in top_face

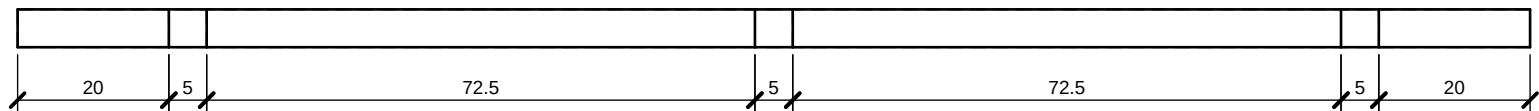
part	vertical
of	shoji_frame
make	3
cut size	300 x 5 x 20
material	oak
scale	1:2, mm



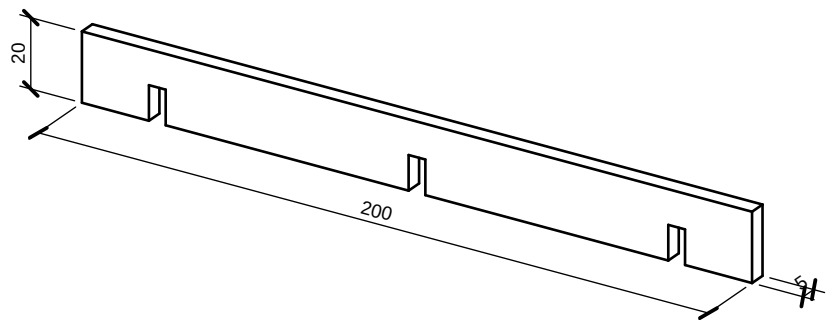
TOP FACE



FRONT EDGE



BOTTOM FACE



ISOMETRIC

3 pieces: horizontal_1, horizontal_2, horizontal_3
 1 3 x dado 5 mm, 10 deep, in bottom_face

part	horizontal
of	shoji_frame
make	3
cut size	200 x 5 x 20
material	oak
scale	1:1, mm

Cut list

vertical	300 x 5 x 20 mm	oak, 3 pieces: vertical_1, vertical_2, vertical_3
	- 3 x dado 5 mm, 10 deep, in top_face	
horizontal	200 x 5 x 20 mm	oak, 3 pieces: horizontal_1, horizontal_2, horizontal_3
	- 3 x dado 5 mm, 10 deep, in bottom_face	

Appendix: source

Made with woodwright 0.9.0, build123d 0.13.0, Python 3.14.4.

To make these plans again:

```
uvx --python 3.14 woodwright==0.9.0 export frames.shoji_frame
```

frames.shoji_frame, in woodwright/examples/frames.py

```
def shoji_frame(
    width: float = 200,
    height: float = 300,
    bar_width: float = 5,
    overhang: float = 20,
    stock: Stock = KUMIKO,
) -> Assembly:
    """A small shoji lattice of three vertical and three horizontal bars,
    crossing in nine cross laps; outside sizes.

    The bars reach `overhang` beyond the outer ones. They are `bar_width`
    wide in the lattice and the stock thick across it; each lap takes half
    of that. Where they cross, the vertical bars keep their front halves and
    the horizontal ones their back halves (the reference faces are at the back).
    """
    asm = Assembly("shoji_frame", root_location=STANDING)
    verticals = [asm.add(Board(f"vertical_{i + 1}", stock, height, bar_width)) for i in range(3)]
    horizontals = [asm.add(Board(f"horizontal_{i + 1}", stock, width, bar_width)) for i in range(3)]

    def starts(length: float) -> list[float]:
        """Where the three bars crossing a bar of this length start along it."""
        return [overhang, (length - bar_width) / 2, length - overhang - bar_width]

    # The horizontal bars run to the right, so their left ends cross each
    # vertical bar from its back edge.
    for vertical, x in zip(verticals, starts(width), strict=True):
        for horizontal, y in zip(horizontals, starts(height), strict=True):
            asm.join(
                CrossLap(
                    vertical.back_edge,
```

```
        horizontal.left_end,  
        position=offset(y, vertical.left_end),  
        overhang=x,  
    )  
    )  
    return asm
```