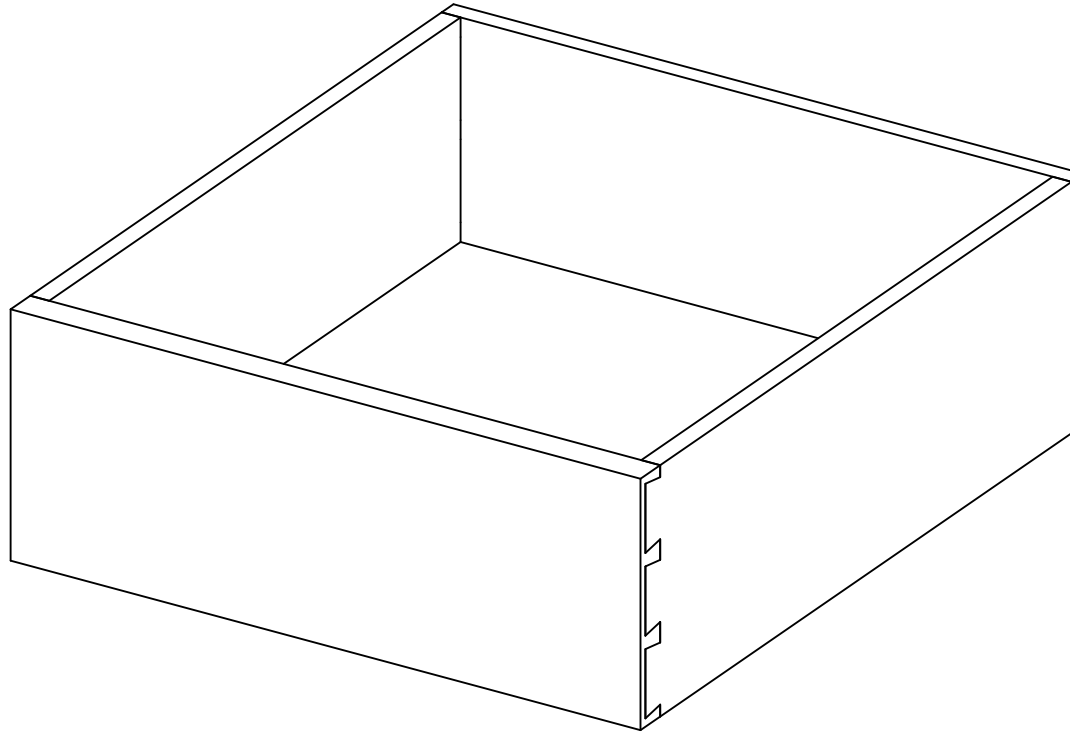


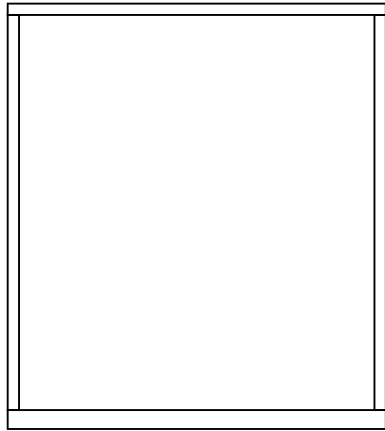
Drawer



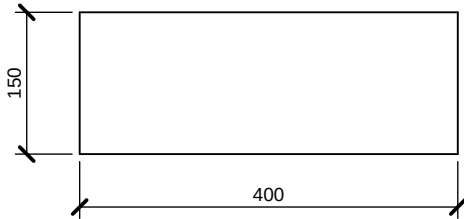
parts	5 (4 different)
overall	400 x 450 x 150 (W x D x H)
material	oak
date	2026-10-25
made with	woodwright 0.9.0

Contents

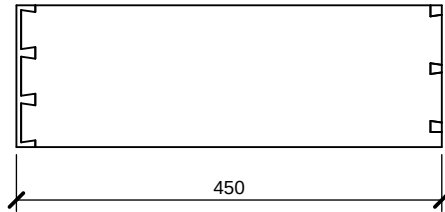
Drawings	3
Assembly	3
Parts	4
1 front	5
2 back	6
3 left_right	7
4 bottom	8
 Cut list	 9
 Appendix: source	 10



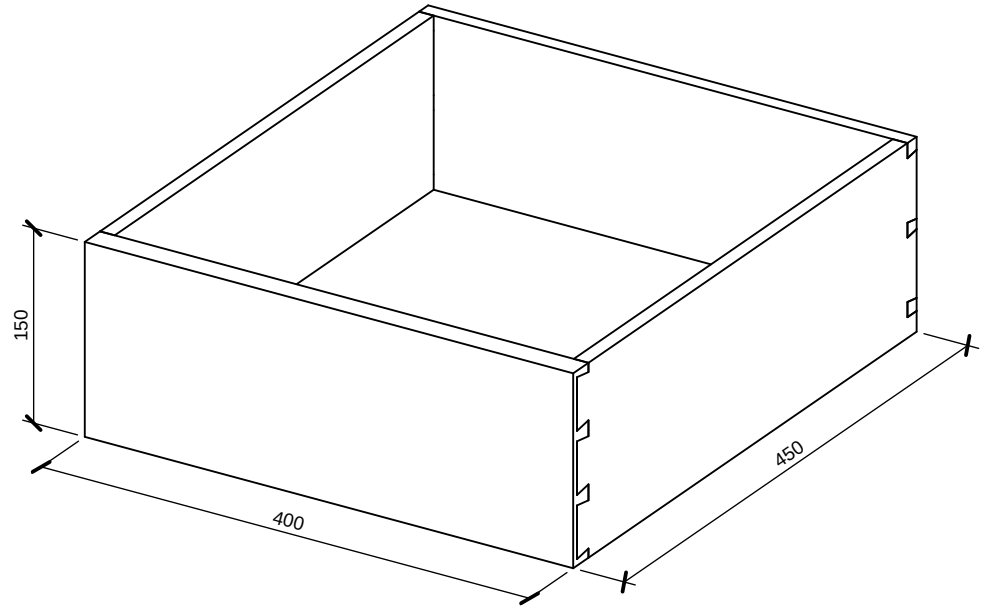
TOP



FRONT

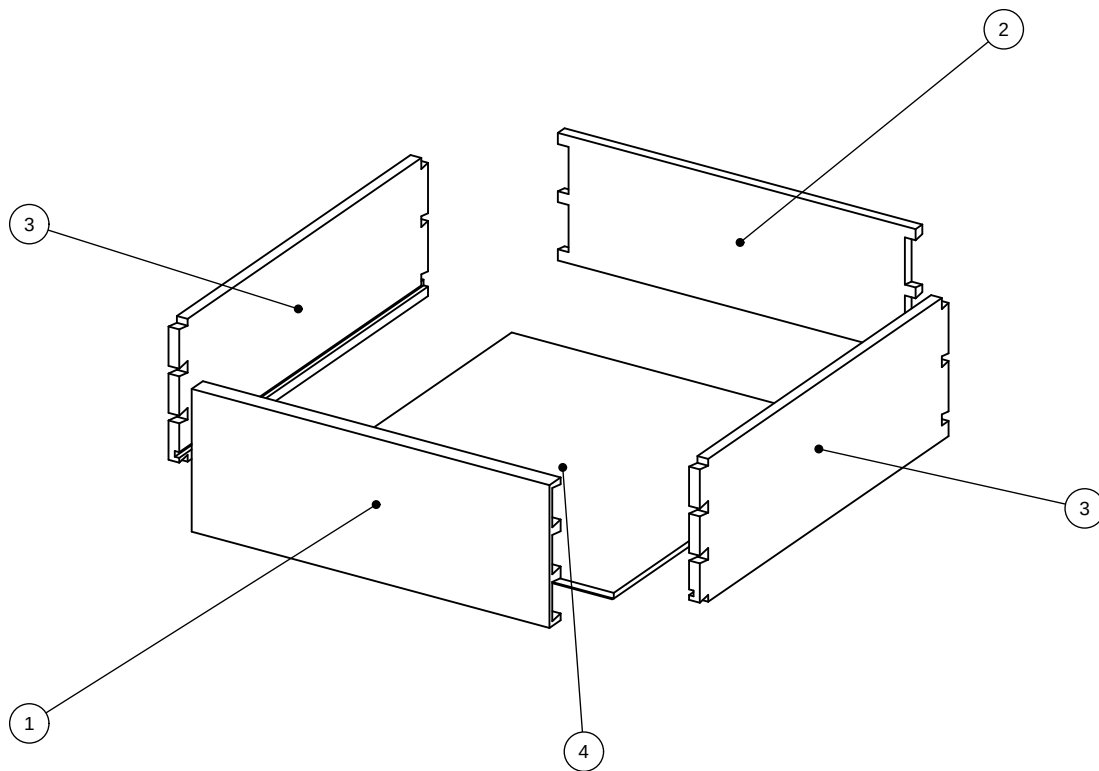


RIGHT SIDE



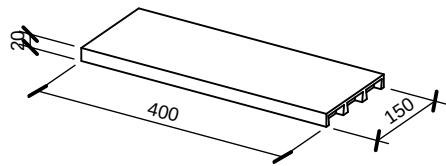
ISOMETRIC

assembly	drawer
parts	5 (4 different)
overall	400 x 450 x 150 (W x D x H)
material	oak
scale	1:8, mm

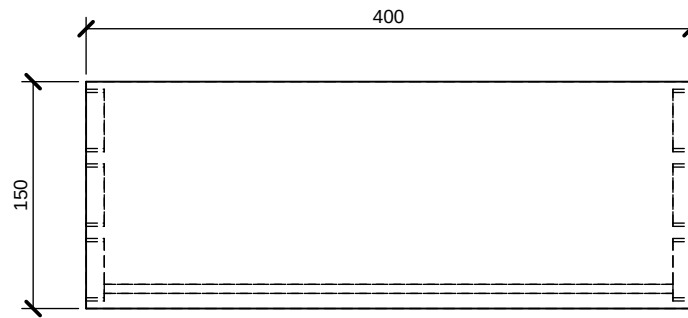


no.	part	qty	cut size (L x W x T)	material	
1	front	1	400 x 150 x 20	oak	
2	back	1	400 x 134 x 12	oak	mirror image
3	left_right	2	445 x 150 x 12	oak	1 mirror image
4	bottom	1	386 x 435 x 10	oak	

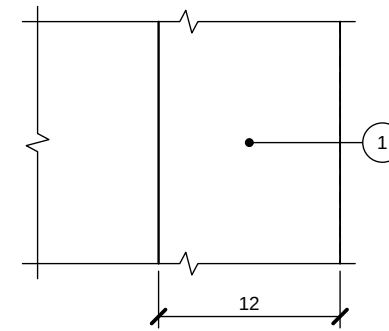
assembly	drawer
parts	5 (4 different)
overall	400 x 450 x 150 (W x D x H)
material	oak
scale	not to scale, mm



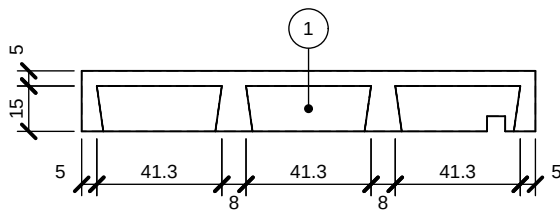
ISOMETRIC



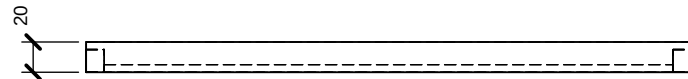
TOP FACE



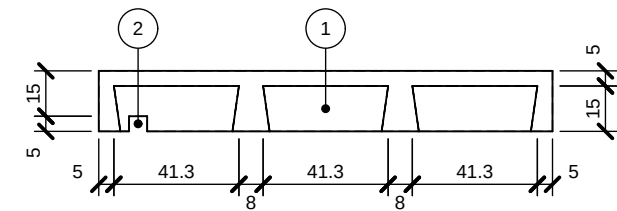
BOTTOM FACE: 1 2:1



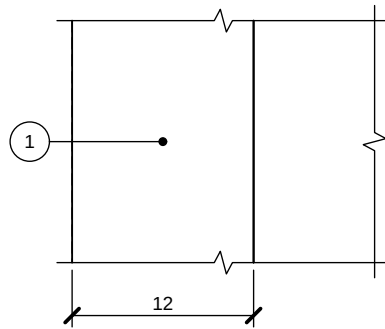
LEFT END 1:2.5



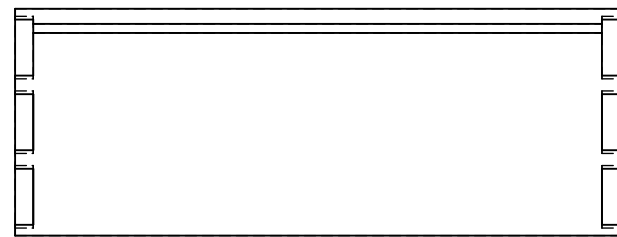
FRONT EDGE



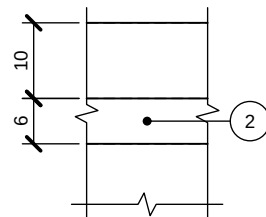
RIGHT END 1:2.5



BOTTOM FACE: 1 2:1



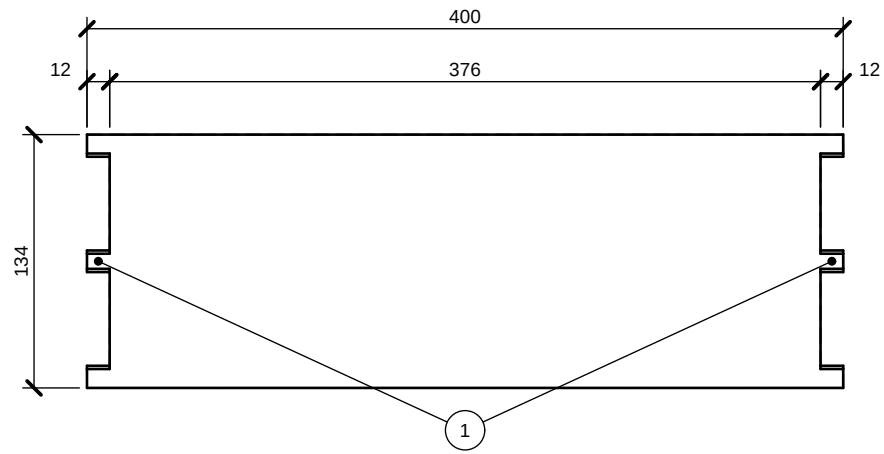
BOTTOM FACE



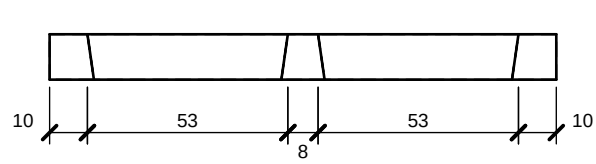
BOTTOM FACE: 2 1:1

- 1 2 x sockets for 3 half-blind dovetails, 15 deep, lap 5
- 2 groove 6 mm, 5 deep, in bottom_face

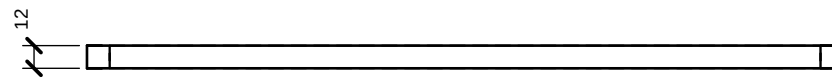
part	front
of	drawer
make	1
cut size	400 x 150 x 20
material	oak
scale	1:5, mm



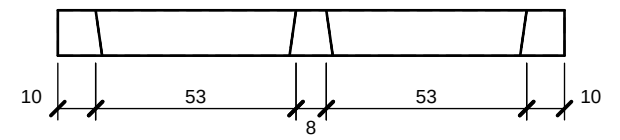
TOP FACE



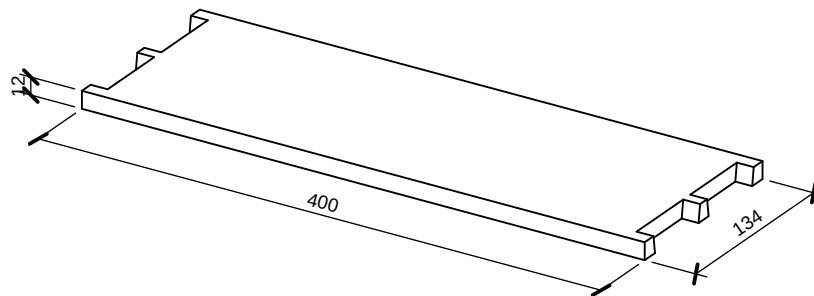
RIGHT END 1:2



FRONT EDGE



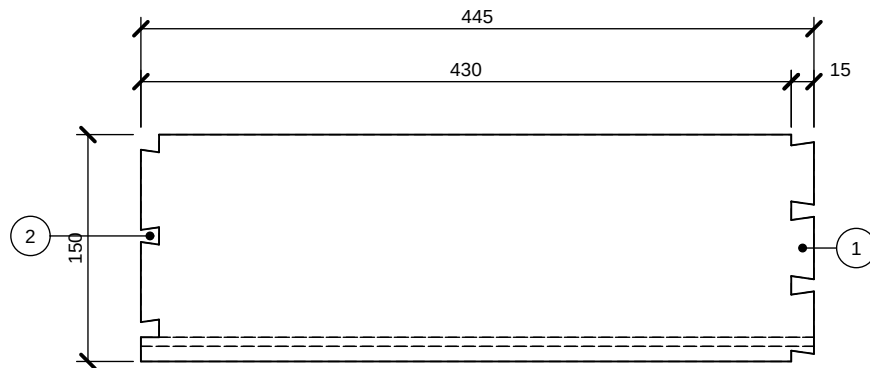
LEFT END 1:2



ISOMETRIC

1 2 x sockets for 2 dovetails

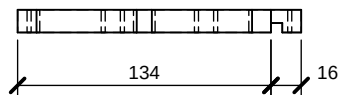
part	back
of	drawer
make	1
cut size	400 x 134 x 12
material	oak, mirror image
scale	1:4, mm



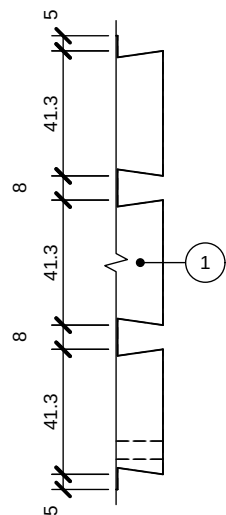
TOP FACE



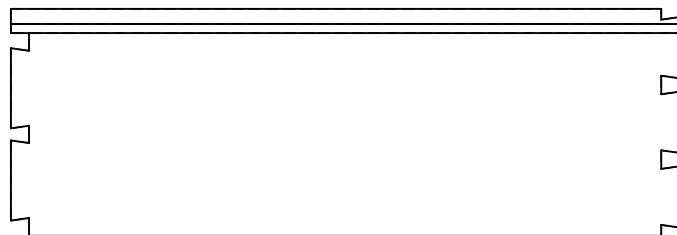
FRONT EDGE



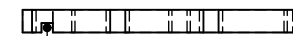
LEFT END 1:4



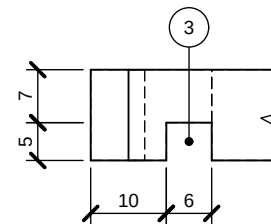
TOP FACE: 1 1:2.5



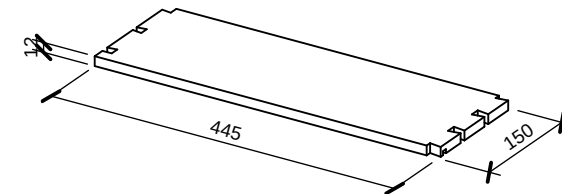
BOTTOM FACE



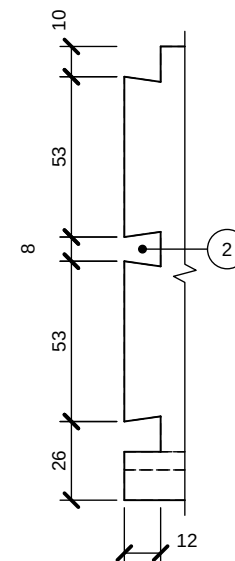
RIGHT END 1:4



RIGHT END: 3 1:1



ISOMETRIC



TOP FACE: 2 1:2.5

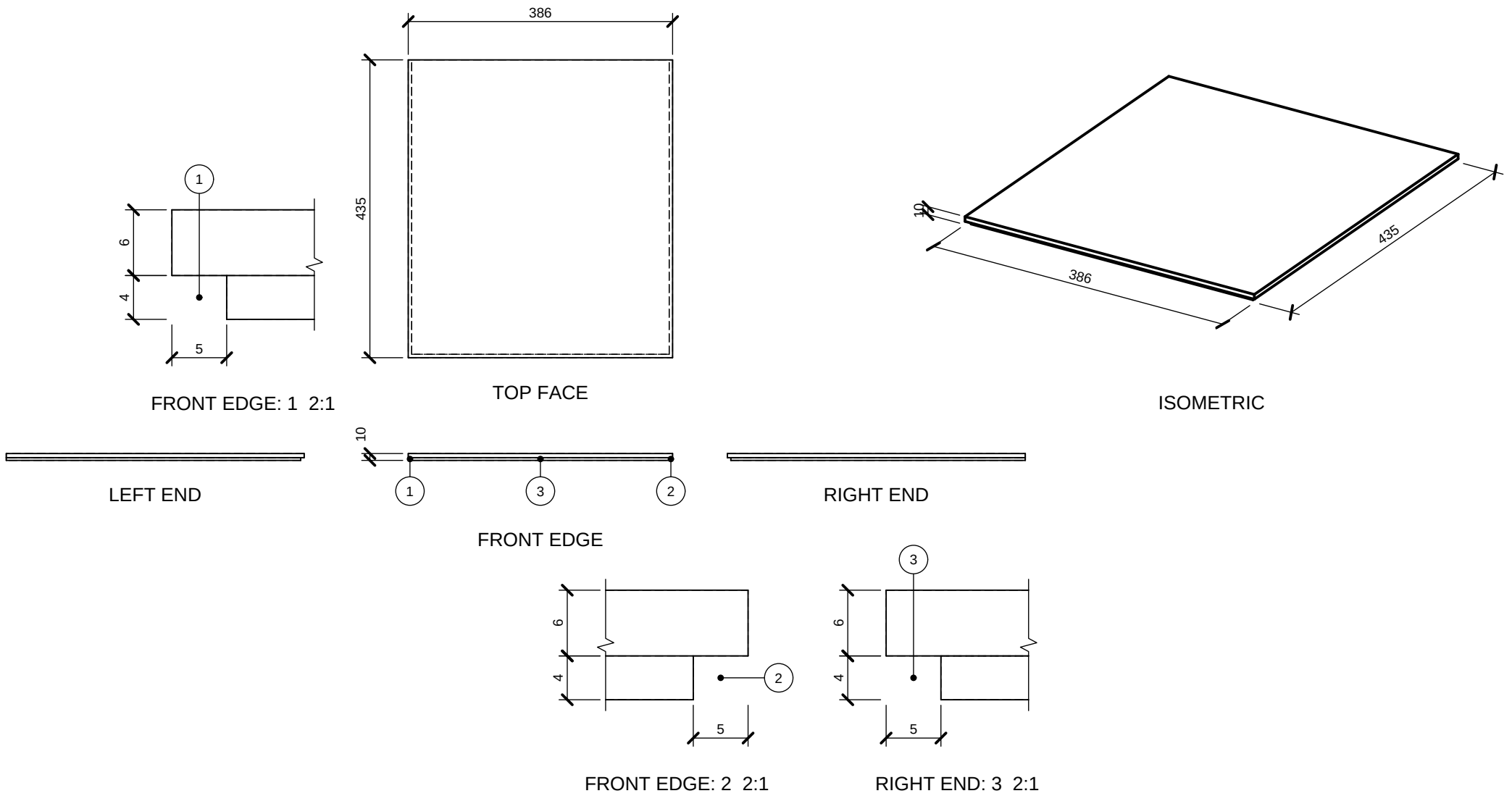
2 pieces: left; mirror image: right

1 3 half-blind dovetails, 15 long, slope 1:7

2 2 dovetails, slope 1:7

3 groove 6 mm, 5 deep, in bottom_face

part	left_right
of	drawer
make	2 (1 mirror image)
cut size	445 x 150 x 12
material	oak
scale	1:5, mm



- 1 end rabbet 5 mm, 4 deep, in bottom_face at left_end
- 2 end rabbet 5 mm, 4 deep, in bottom_face at right_end
- 3 rabbet 5 mm, 4 deep, in bottom_face at front_edge

part	bottom
of	drawer
make	1
cut size	386 x 435 x 10
material	oak
scale	1:8, mm

Cut list

front	400 x 150 x 20 mm oak
	- 2 x sockets for 3 half-blind dovetails, 15 deep, lap 5
	- groove 6 mm, 5 deep, in bottom_face
back	400 x 134 x 12 mm oak, mirror image
	- 2 x sockets for 2 dovetails
left_right	445 x 150 x 12 mm oak, 2 pieces: left; mirror image: right
	- 3 half-blind dovetails, 15 long, slope 1:7
	- 2 dovetails, slope 1:7
	- groove 6 mm, 5 deep, in bottom_face
bottom	386 x 435 x 10 mm oak
	- end rabbet 5 mm, 4 deep, in bottom_face at left_end
	- end rabbet 5 mm, 4 deep, in bottom_face at right_end
	- rabbet 5 mm, 4 deep, in bottom_face at front_edge

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 projects.drawer
```

projects.drawer, in woodwright/examples/projects.py

```
def drawer(
    width: float = 400,
    depth: float = 450,
    height: float = 150,
    groove_height: float = 10,
    groove_width: float = 6,
    groove_depth: float = 5,
    stock: Stock = OAK,
    side_stock: Stock = SIDE_OAK,
    bottom_stock: Stock = THIN_OAK,
) -> Assembly:
    """A classic drawer with a bottom; outside sizes.

    The front has half-blind dovetails, so the tails don't show from the
    front; its bottom half pins are narrow, so that the groove runs through
    the bottom tails and is hidden. The back is through dovetailed. The bottom slides in grooves in
    the sides and the front, under the back, which is narrower and sits on
    it. The bottom's edges are rabbeted on its underside to fit the grooves,
    its top flush with their tops; its grain runs from side to side, so it
    moves towards the back. The groove's lower wall is `groove_height` above
    the bottom edges. The back and the right side are mirror images of the
    front and the left side, all with their reference faces outside.
    """
    s, b = side_stock.thickness, bottom_stock.thickness
    side_length = depth - stock.thickness - s
    asm = Assembly("drawer", root_location=BOX_FRONT)
    front = asm.add(Board("front", stock, length=width - 2 * s, width=height))
    back_height = height - groove_height - groove_width
    back = asm.add(Board("back", side_stock, length=width - 2 * s, width=back_height, mirror=True))
    left = asm.add(Board("left", side_stock, length=side_length, width=height))
```

```

right = asm.add(Board("right", side_stock, length=side_length, width=height, mirror=True))
bottom = asm.add(Board("bottom", bottom_stock, length=width - 2 * s, width=side_length + s))

def in_groove(tongue: SideRef, groove: SideRef, position: Position) -> TongueAndGroove:
    """The bottom's edge or end in a groove in the inside face of the front or a side."""
    return TongueAndGroove(
        tongue=tongue,
        groove=groove,
        tongue_thickness=groove_width,
        tongue_length=groove_depth,
        tongue_offset=0,
        position=position,
        flush=(groove.part.ref_edge, bottom.bottom_face),
        across=offset(groove_height + groove_width - b),
    )

# The groove runs out of the front's ends: it must pass through the bottom
# tail, not through the half pin below it, which shows on the side. The
# half pin is narrowest at the lap and widens by the slope towards the
# inside (by 1:7); the groove stays at least 2 mm above it.
socket_depth = stock.thickness - rules.dovetail_lap(stock.thickness)
half_pin = math.floor(groove_height - socket_depth / 7 - 2)
# The sides' right ends are at the front; the back's top edge is flush with theirs.
for side, front_end, back_end, bottom_end in [
    (left, front.left_end, back.left_end, bottom.left_end),
    (right, front.right_end, back.right_end, bottom.right_end),
]:
    asm.join(
        HalfBlindDovetail(
            tails=side.right_end, pins=front_end, tail_count=3, half_pin_width=half_pin
        )
    )
    asm.join(
        ThroughDovetail(
            tails=side.left_end,
            pins=back_end,
            tail_count=2,
            flush=(back.back_edge, side.back_edge),
        )
    )
    asm.join(in_groove(bottom_end, side.bottom_face, flush_with(side.right_end)))

```

```
asm.join(in_groove(bottom.front_edge, front.bottom_face, flush_with(front.left_end)))  
# The back sits on the bottom, which reaches to its outside face.  
asm.join(Butt(bottom.top_face, back.ref_edge, flush=(bottom.back_edge, back.ref_face)))  
return asm
```