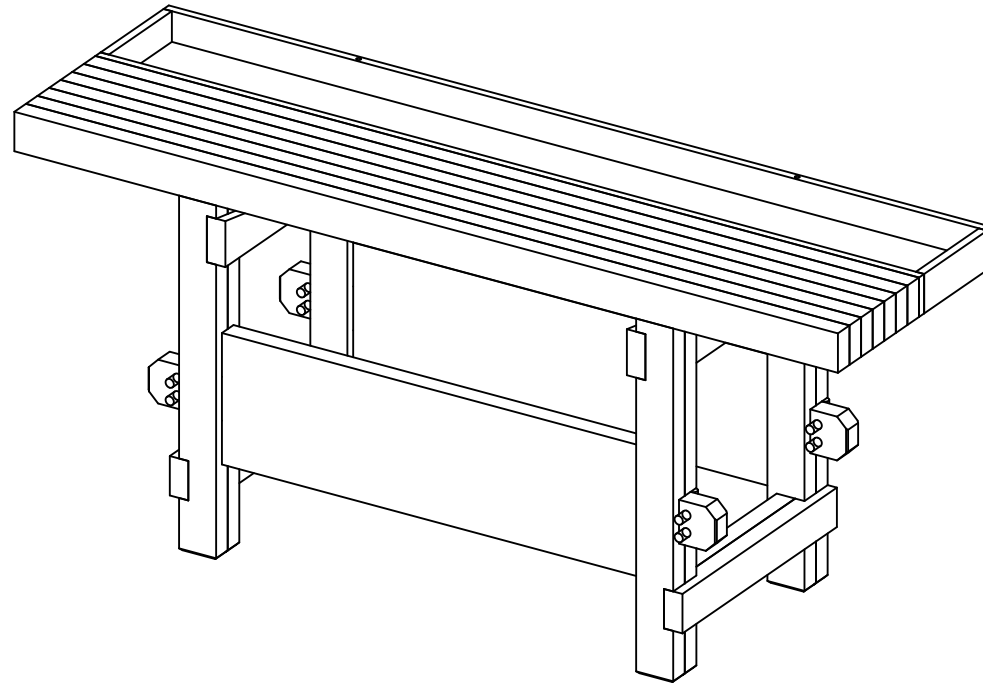


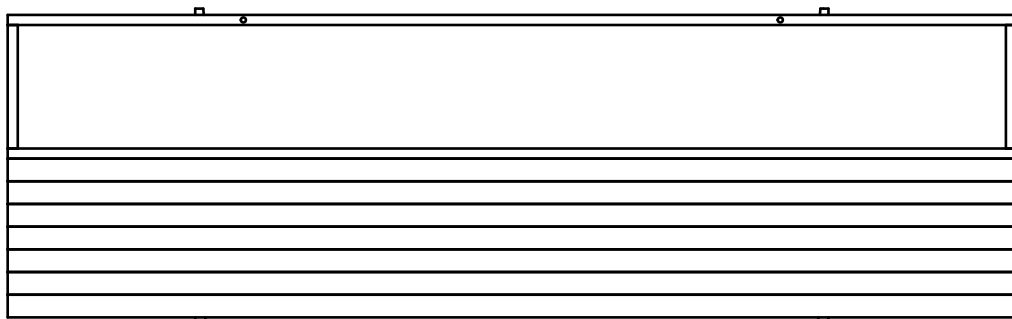
Workbench



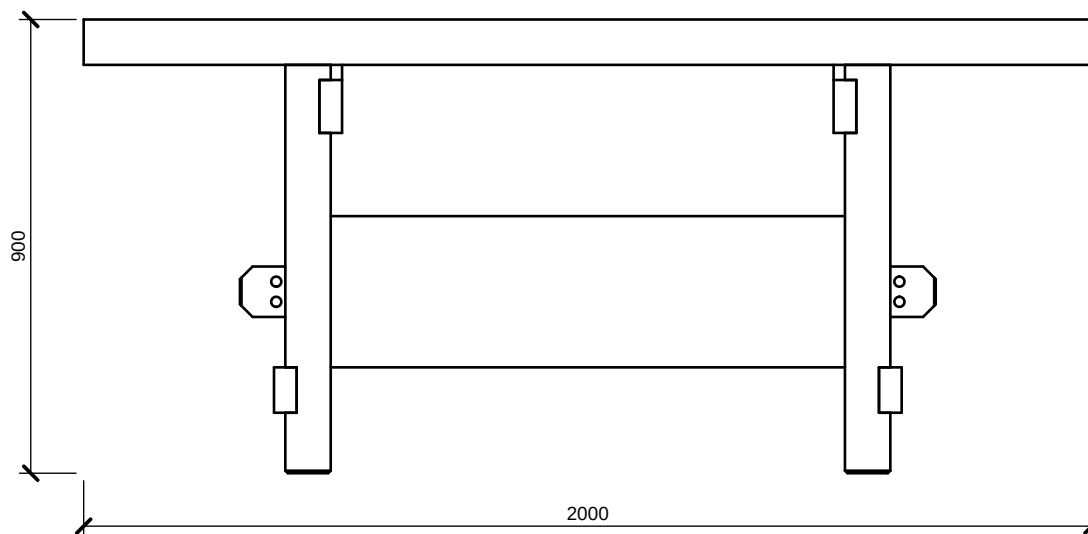
parts	40 (13 different)
overall	2000 x 620 x 900 (W x D x H)
material	oak
date	2026-10-25
made with	woodwright 0.9.0

Contents

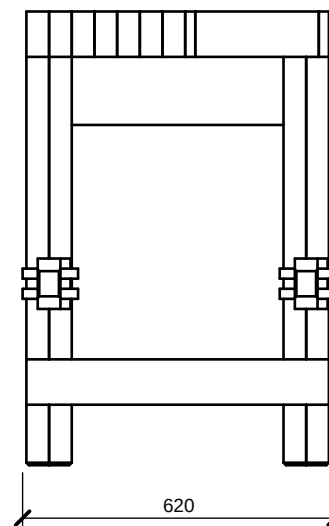
Drawings	3
Assembly	3
Parts	4
1 leg	5
2 lower_rail	6
3 upper_rail	7
4 stretcher	8
5 tusk	9
6 bearing	10
7 top_beam_1_2_3_5_6	11
8 top_beam_4_7	12
9 tray_front_back	13
10 tray_left_right	14
11 tray_bottom	15
12 top_beam_dowel	16
13 tray_front_dowel	17
Cut list	18
Appendix: source	19



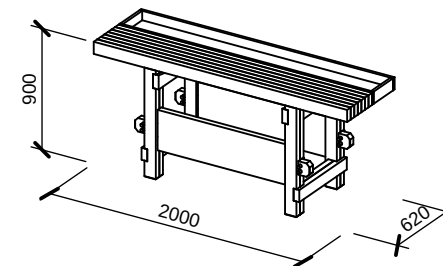
TOP



FRONT

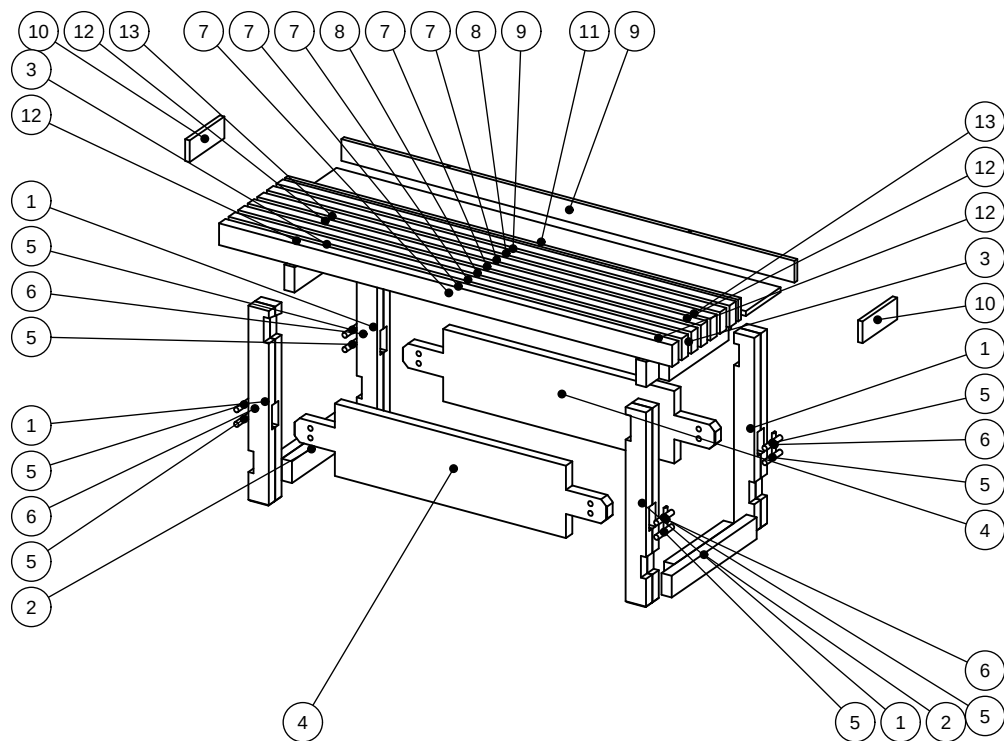


RIGHT SIDE



ISOMETRIC

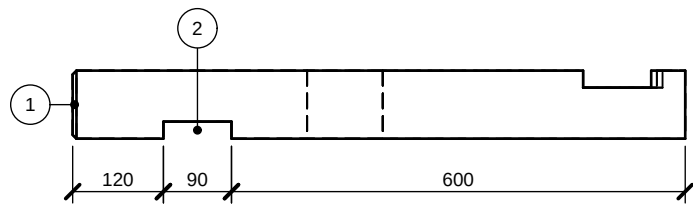
assembly	workbench
parts	40 (13 different)
overall	2000 x 620 x 900 (W x D x H)
material	oak
scale	1:15, mm



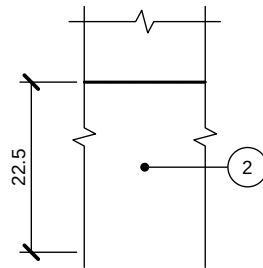
EXPLODED

no.	part	qty	cut size (L x W x T)	material	
1	leg	4	810 x 90 x 90	oak	2 mirror image
2	lower_rail	2	600 x 90 x 90	oak	
3	upper_rail	2	600 x 135 x 90	oak	1 mirror image
4	stretcher	2	1380 x 300 x 45	oak	
5	tusk	8	∅20 x 110	oak	
6	bearing	4	∅20 x 100	oak	
7	top_beam_1_2_3_5_6	5	2000 x 90 x 45	oak	
8	top_beam_4_7	2	2000 x 90 x 45	oak	
9	tray_front_back	2	2000 x 90 x 20	oak	
10	tray_left_right	2	245 x 90 x 20	oak	
11	tray_bottom	1	1960 x 245 x 20	oak	
12	top_beam_dowel	4	∅20 x 50	oak	
13	tray_front_dowel	2	∅10 x 40	oak	

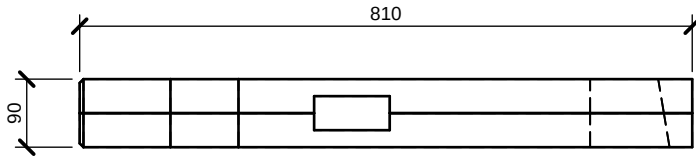
assembly	workbench
parts	40 (13 different)
overall	2000 x 620 x 900 (W x D x H)
material	oak
scale	not to scale, mm



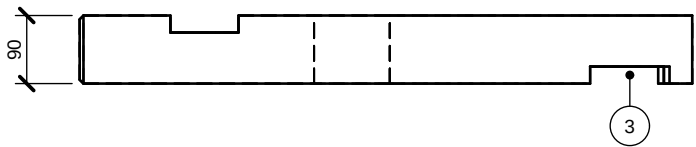
BACK EDGE



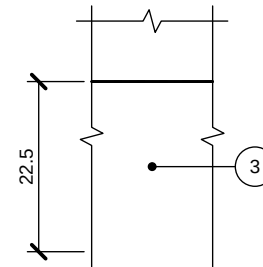
BACK EDGE: 2 1:1



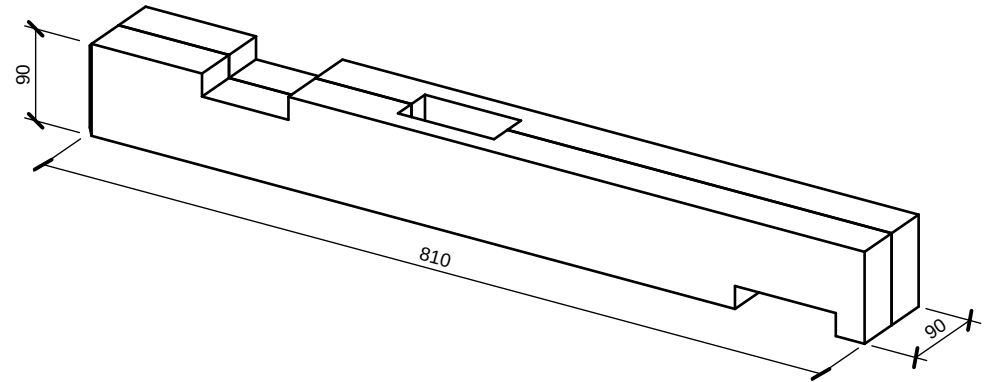
TOP FACE



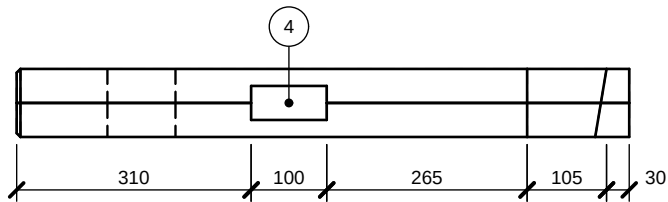
FRONT EDGE



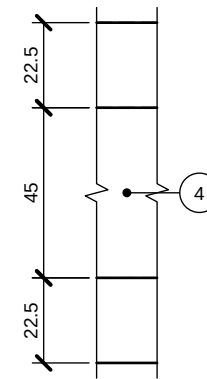
FRONT EDGE: 3 1:1



ISOMETRIC



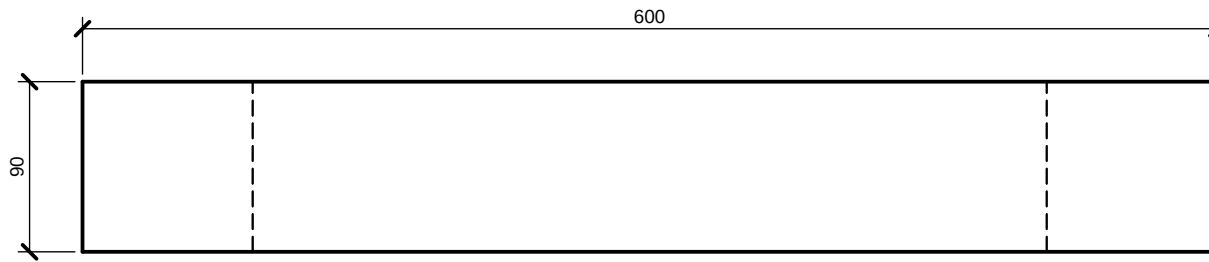
BOTTOM FACE



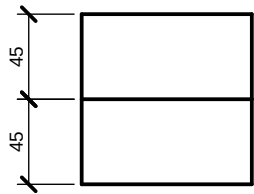
BOTTOM FACE: 4 1:2

- 4 pieces: left_front_leg, right_back_leg; mirror image: left_back_leg, right_front_leg
- 1 chamfer 5 mm around left_end
 - 2 dado 90 mm, 22.5 deep, in top_face
 - 3 dovetail socket 90 to 105 mm, 22.5 deep, 1:6, in bottom_face, 30 mm short of right_end
 - 4 through mortise 100 x 45 mm in bottom_face

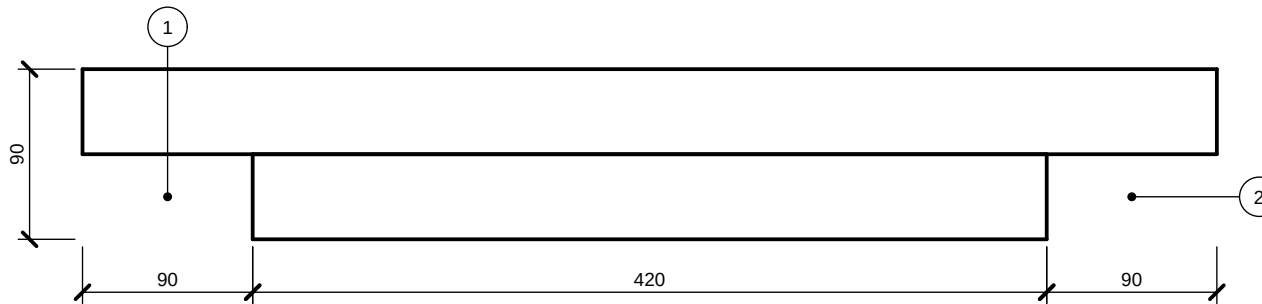
part	leg
of	workbench
make	4 (2 mirror image)
cut size	810 x 90 x 90
material	oak
scale	1:10, mm



TOP FACE



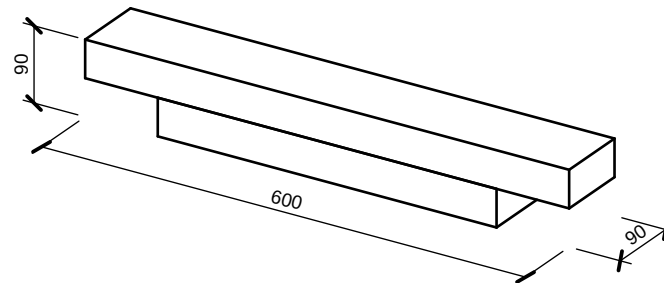
LEFT END



FRONT EDGE



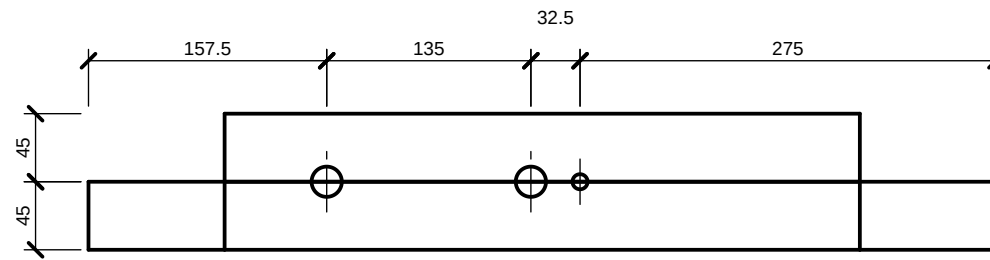
RIGHT END



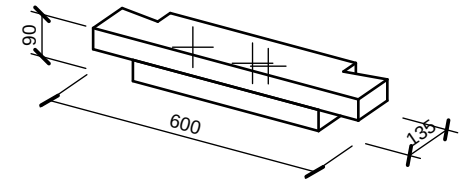
ISOMETRIC

2 pieces: left_lower_rail, right_lower_rail
 1 lap 45 mm thick, 90 long, on left_end
 2 lap 45 mm thick, 90 long, on right_end

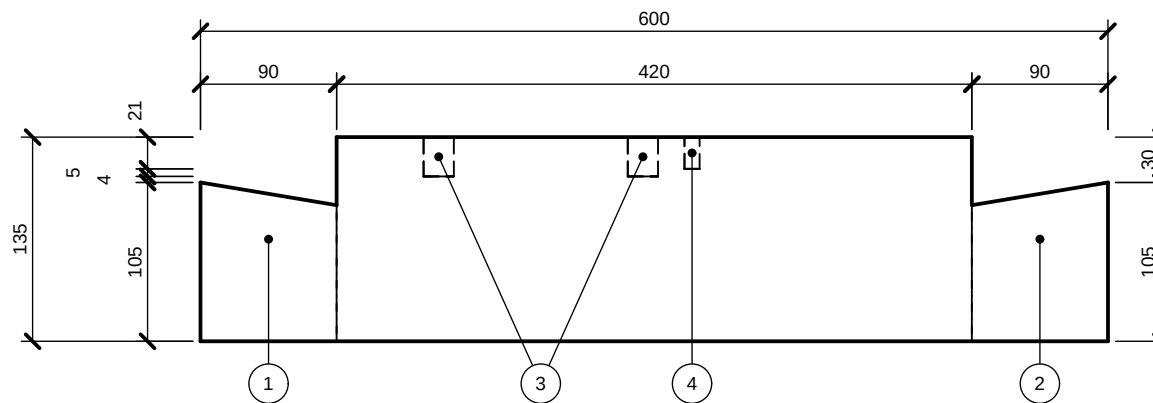
part	lower_rail
of	workbench
make	2
cut size	600 x 90 x 90
material	oak
scale	1:4, mm



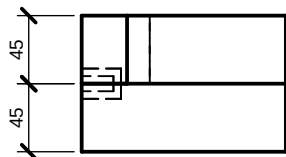
BACK EDGE



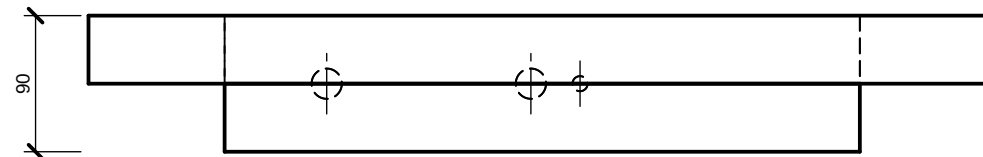
ISOMETRIC



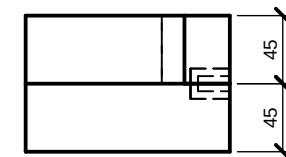
TOP FACE



LEFT END



FRONT EDGE



RIGHT END

2 pieces: left_upper_rail; mirror image: right_upper_rail

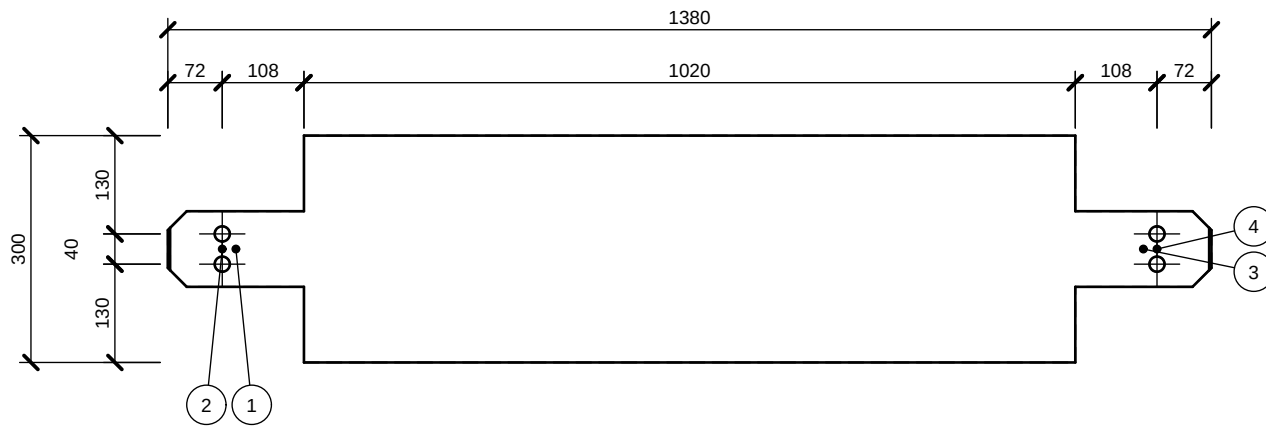
1 half-dovetail lap 45 mm thick, 90 long, 90 wide at the shoulder, 1:6, cut back 30 at back_edge, on left_end

2 half-dovetail lap 45 mm thick, 90 long, 90 wide at the shoulder, 1:6, cut back 30 at back_edge, on right_end

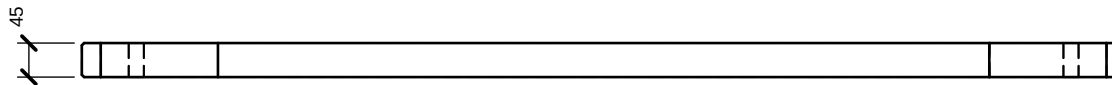
3 2 x dowel hole $\varnothing 20$, 26 deep, in back_edge

4 dowel hole $\varnothing 10$, 21 deep, in back_edge

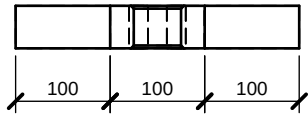
part	upper_rail
of	workbench
make	2 (1 mirror image)
cut size	600 x 135 x 90
material	oak
scale	1:5, mm



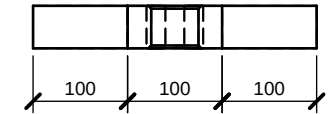
TOP FACE



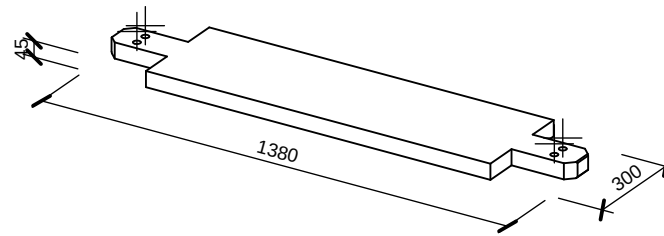
FRONT EDGE



LEFT END 1:8



RIGHT END 1:8

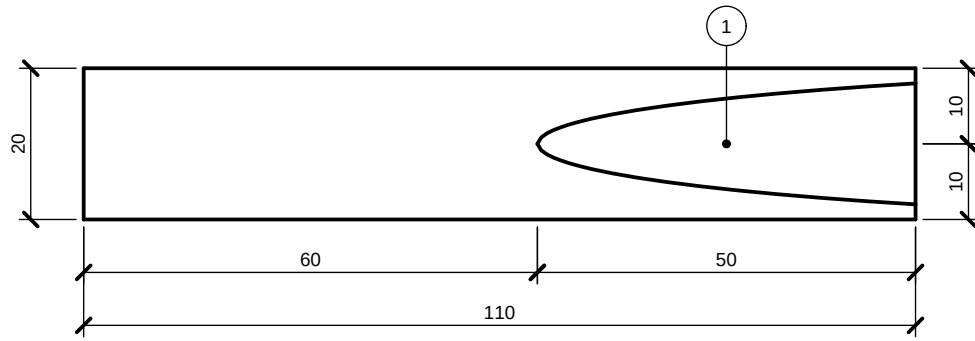


ISOMETRIC

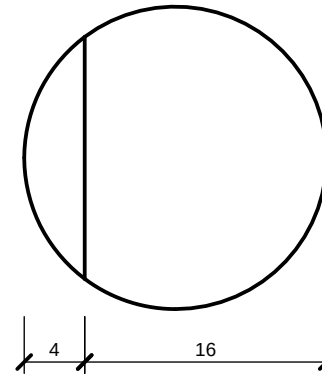
2 pieces: front_stretcher, back_stretcher

- 1 tenon 100 x 45 mm, 180 long, end chamfered 3 mm, corners chamfered 25 mm, on left_end
- 2 2 holes $\varnothing 20$ for tusks through the tenon on left_end, parallel to left_front_leg.top_face, 8 mm off it
- 3 tenon 100 x 45 mm, 180 long, end chamfered 3 mm, corners chamfered 25 mm, on right_end
- 4 2 holes $\varnothing 20$ for tusks through the tenon on right_end, parallel to right_front_leg.top_face, 8 mm off it

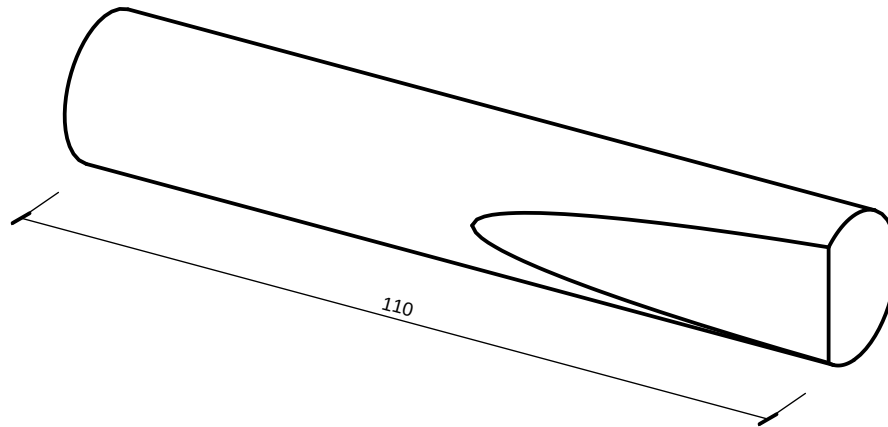
part	stretcher
of	workbench
make	2
cut size	1380 x 300 x 45
material	oak
scale	1:10, mm



SIDE



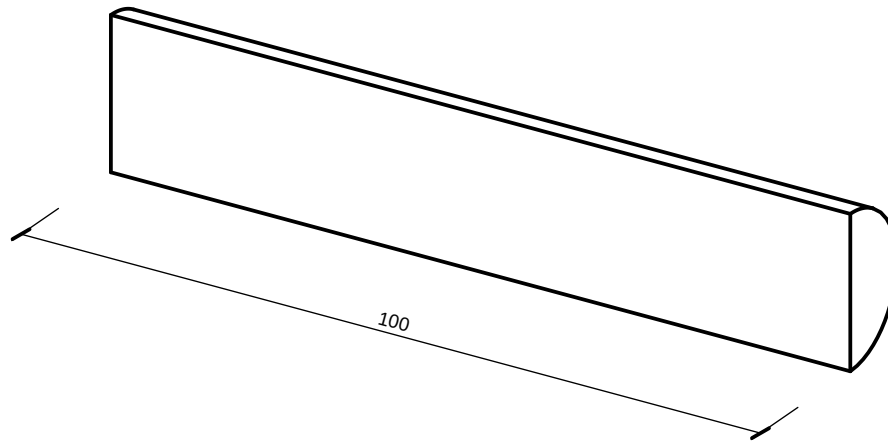
RIGHT END 2:1



ISOMETRIC

8 pieces: front_left_tusk_1, front_left_tusk_2, front_right_tusk_1, front_right_tusk_2, back_left_tusk_1, back_left_tusk_2, back_right_tusk_1, back_right_tusk_2
 1 flat taper 4 mm over 50 on front_edge towards right_end

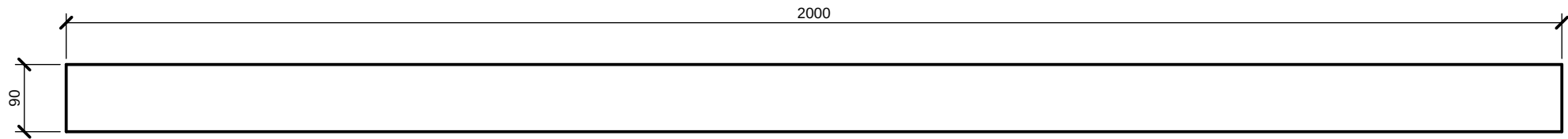
part	tusk
of	workbench
make	8
cut size	ø20 x 110
material	oak
scale	1:1, mm



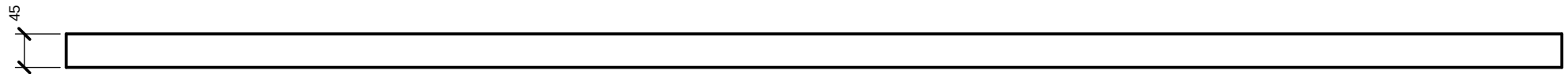
ISOMETRIC

4 pieces: front_left_bearing, front_right_bearing, back_left_bearing, back_right_bearing
 1 flat 20 mm wide, 10 deep, on front_edge

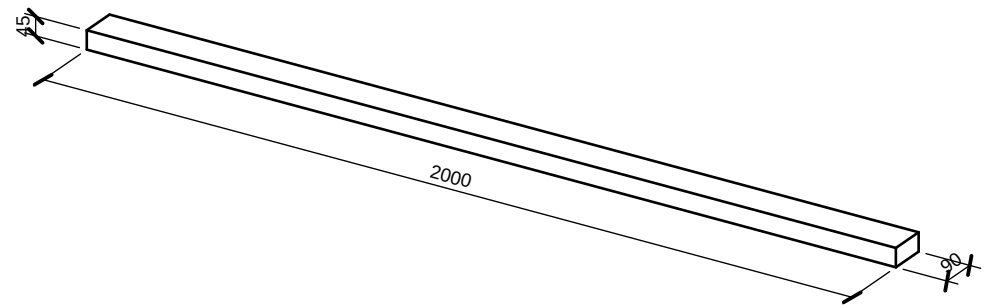
part	bearing
of	workbench
make	4
cut size	ø20 x 100
material	oak
scale	2:1, mm



TOP FACE



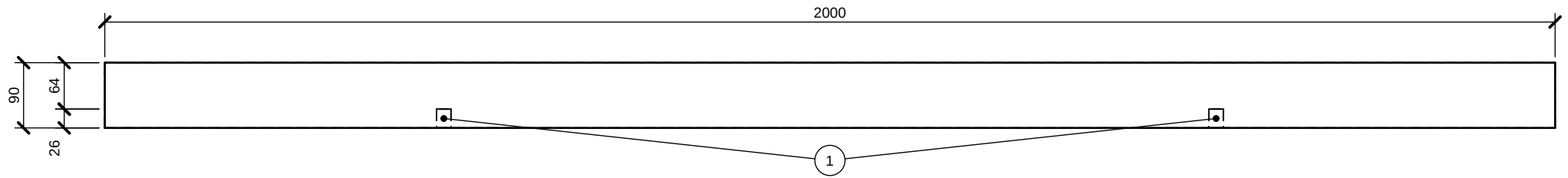
FRONT EDGE



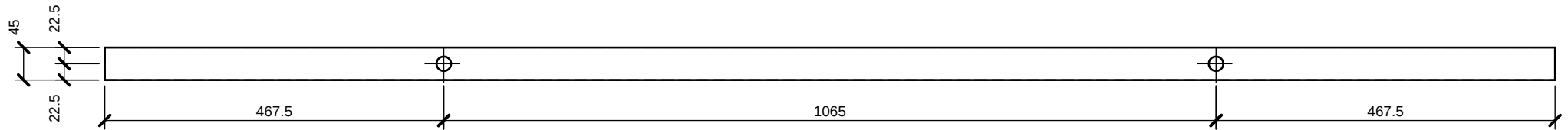
ISOMETRIC

5 pieces: top_beam_1, top_beam_2, top_beam_3, top_beam_5, top_beam_6

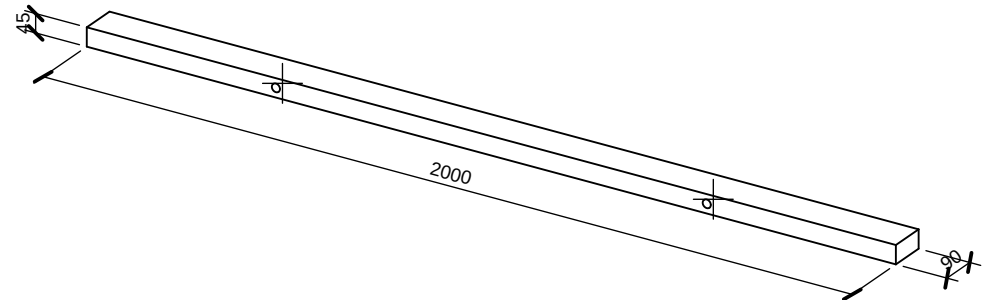
part	top_beam_1_2_3_5_6
of	workbench
make	5
cut size	2000 x 90 x 45
material	oak
scale	1:8, mm



TOP FACE



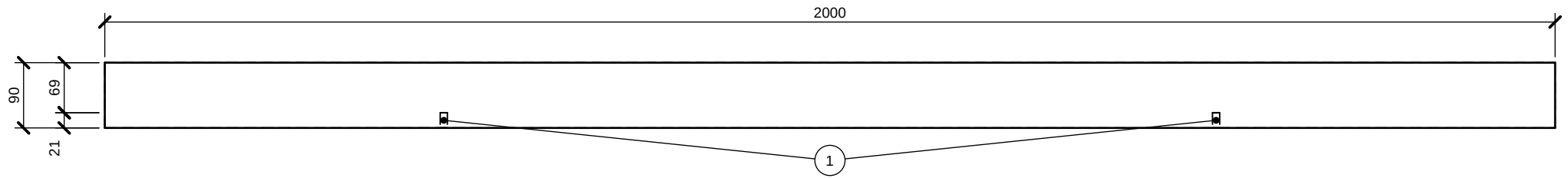
FRONT EDGE



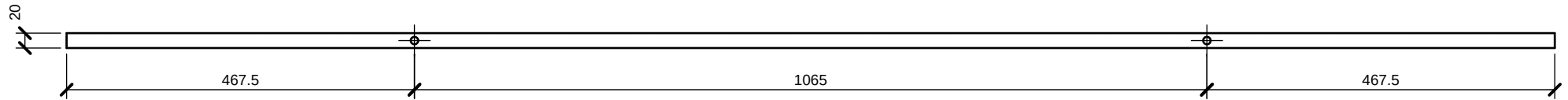
ISOMETRIC

2 pieces: top_beam_4, top_beam_7
 1 2 x dowel hole $\varnothing 20$, 26 deep, in front_edge

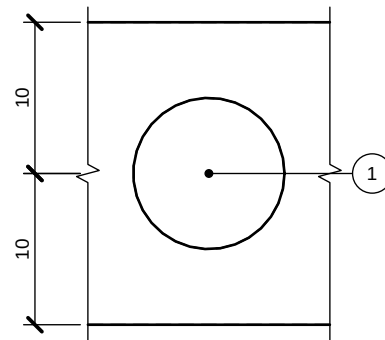
part	top_beam_4_7
of	workbench
make	2
cut size	2000 x 90 x 45
material	oak
scale	1:8, mm



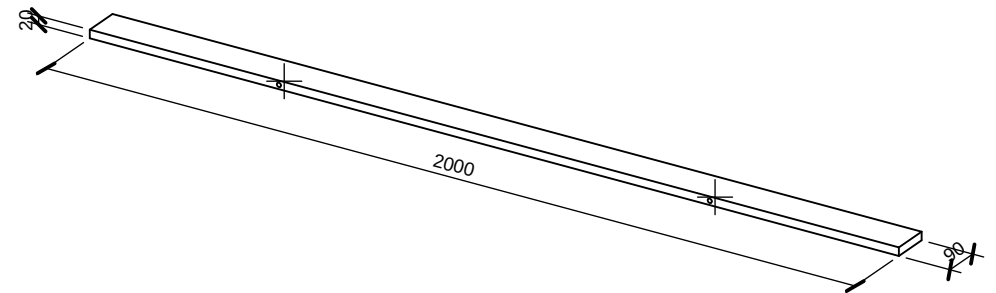
TOP FACE



FRONT EDGE



FRONT EDGE: 1 2:1

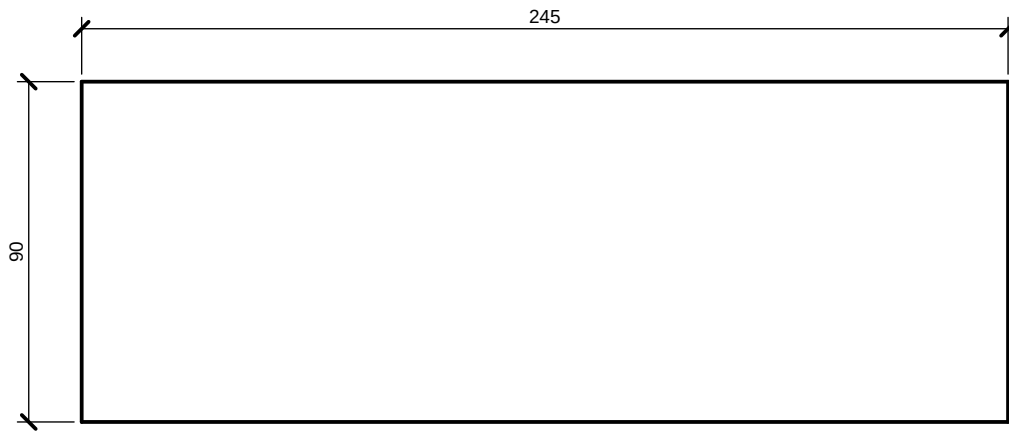


ISOMETRIC

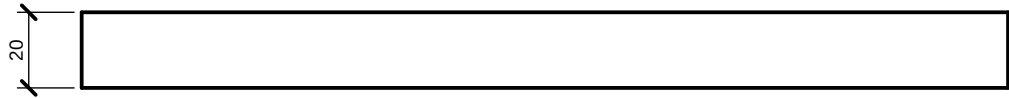
2 pieces: tray_front, tray_back

1 2 x dowel hole $\varnothing 10$, 21 deep, in front_edge

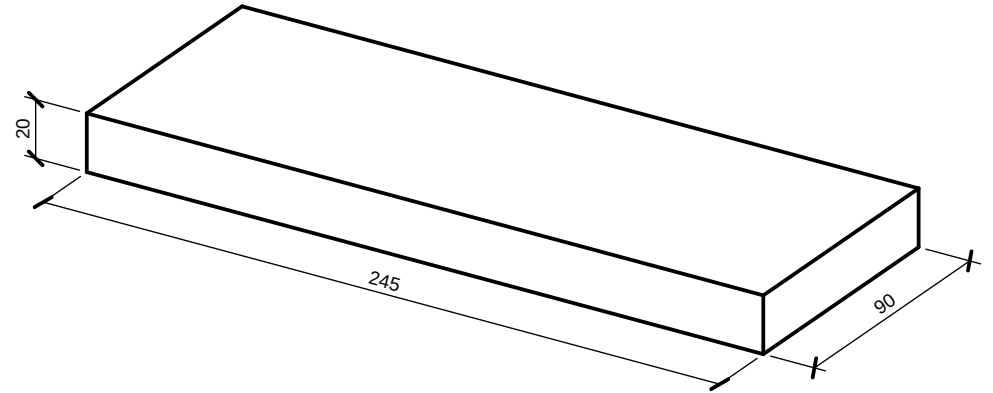
part	tray_front_back
of	workbench
make	2
cut size	2000 x 90 x 20
material	oak
scale	1:8, mm



TOP FACE



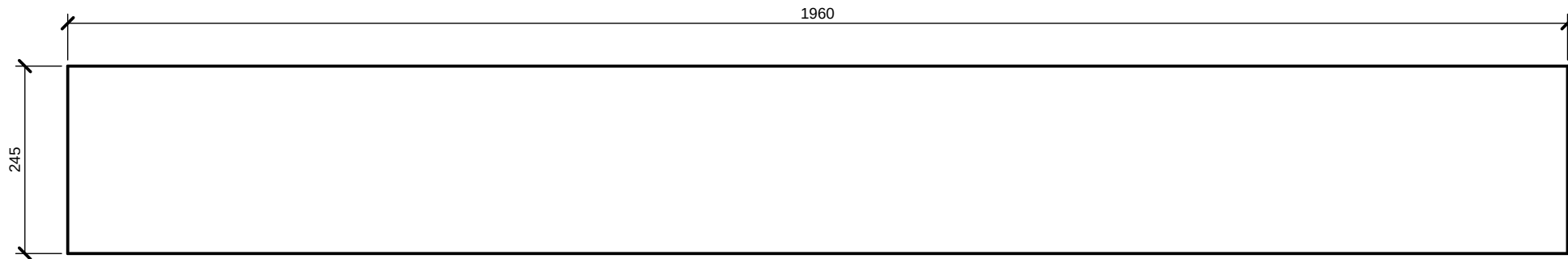
FRONT EDGE



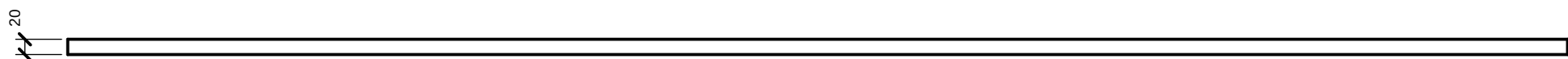
ISOMETRIC

2 pieces: tray_left, tray_right

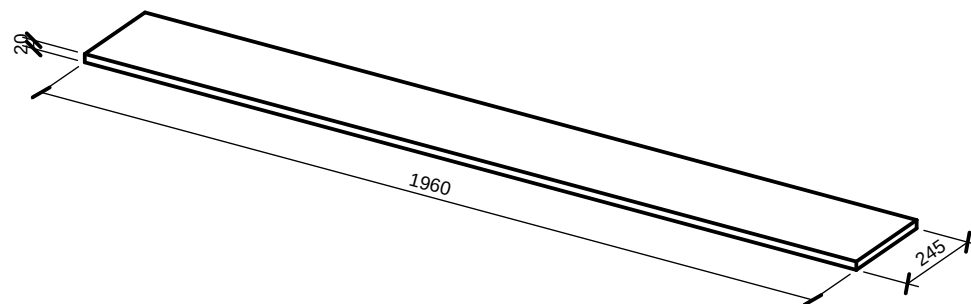
part	tray_left_right
of	workbench
make	2
cut size	245 x 90 x 20
material	oak
scale	1:2, mm



TOP FACE

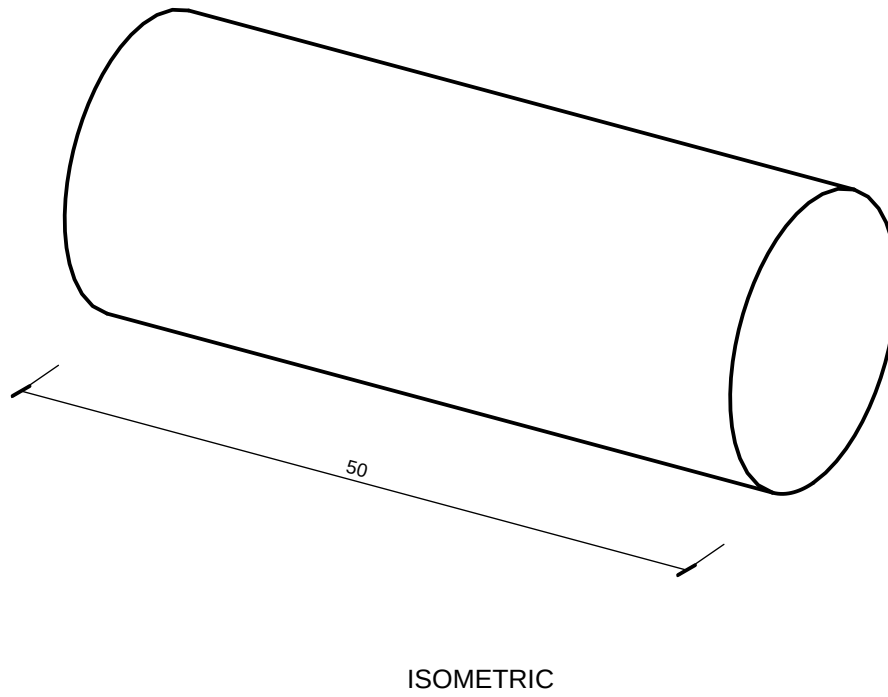
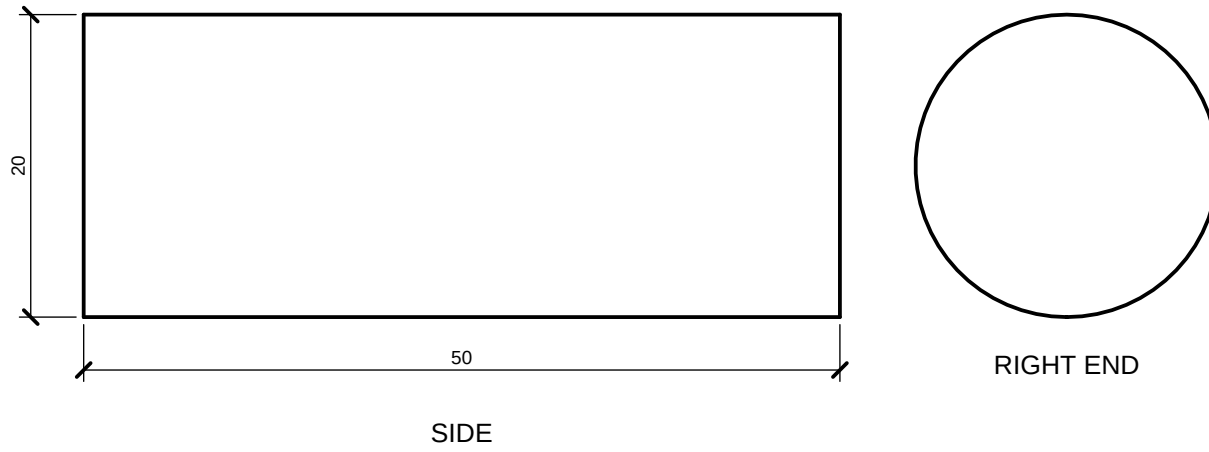


FRONT EDGE



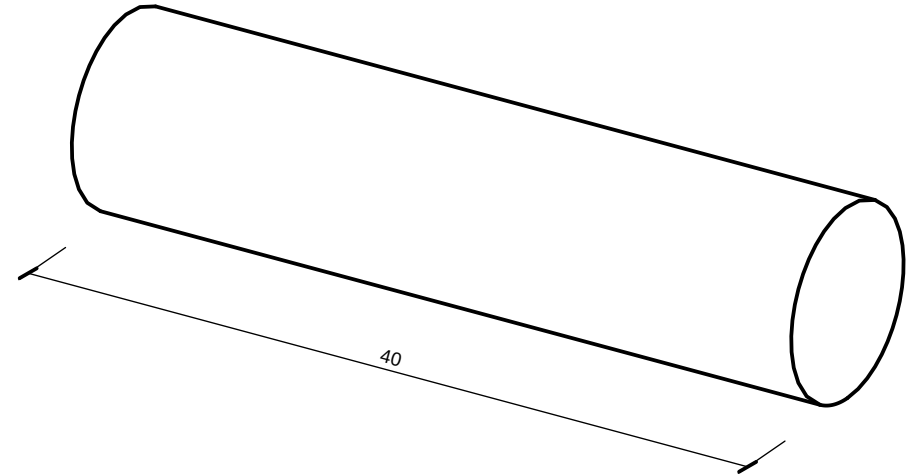
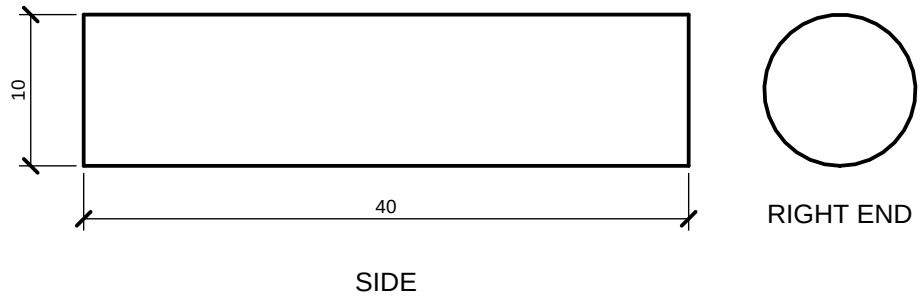
ISOMETRIC

part	tray_bottom
of	workbench
make	1
cut size	1960 x 245 x 20
material	oak
scale	1:8, mm



4 pieces: top_beam_4_dowel_left, top_beam_7_dowel_left, top_beam_4_dowel_right, top_beam_7_dowel_right

part	top_beam_dowel
of	workbench
make	4
cut size	ø20 x 50
material	oak
scale	2:1, mm



ISOMETRIC

2 pieces: tray_front_dowel_left, tray_front_dowel_right

part	tray_front_dowel
of	workbench
make	2
cut size	ø10 x 40
material	oak
scale	2:1, mm

Cut list

leg	810 x 90 x 90 mm oak, glued up from 2 boards 810 x 90 x 45 mm, 4 pieces: left_front_leg, right_back_leg; mirror image: left_back_leg, right_front_leg - chamfer 5 mm around left_end - dado 90 mm, 22.5 deep, in top_face - dovetail socket 90 to 105 mm, 22.5 deep, 1:6, in bottom_face, 30 mm short of right_end - through mortise 100 x 45 mm in bottom_face
lower_rail	600 x 90 x 90 mm oak, glued up from 2 boards 600 x 90 x 45 mm, 2 pieces: left_lower_rail, right_lower_rail - lap 45 mm thick, 90 long, on left_end - lap 45 mm thick, 90 long, on right_end
upper_rail	600 x 135 x 90 mm oak, glued up from 2 boards 600 x 135 x 45 mm, 2 pieces: left_upper_rail; mirror image: right_upper_rail - half-dovetail lap 45 mm thick, 90 long, 90 wide at the shoulder, 1:6, cut back 30 at back_edge, on left_end - half-dovetail lap 45 mm thick, 90 long, 90 wide at the shoulder, 1:6, cut back 30 at back_edge, on right_end - 2 x dowel hole ø20, 26 deep, in back_edge, 67.5 and 202.5 from left_end; centered - dowel hole ø10, 21 deep, in back_edge, 235 from left_end, centered
stretcher	1380 x 300 x 45 mm oak, 2 pieces: front_stretcher, back_stretcher - tenon 100 x 45 mm, 180 long, end chamfered 3 mm, corners chamfered 25 mm, on left_end - 2 holes ø20 for tusks through the tenon on left_end, parallel to left_front_leg.top_face, 8 mm off it - tenon 100 x 45 mm, 180 long, end chamfered 3 mm, corners chamfered 25 mm, on right_end - 2 holes ø20 for tusks through the tenon on right_end, parallel to right_front_leg.top_face, 8 mm off it
tusk	ø20 x 110 mm oak, 8 pieces: front_left_tusk_1, front_left_tusk_2, front_right_tusk_1, front_right_tusk_2, back_left_tusk_1, back_left_tusk_2, back_right_tusk_1, back_right_tusk_2 - flat taper 4 mm over 50 on front_edge towards right_end
bearing	ø20 x 100 mm oak, 4 pieces: front_left_bearing, front_right_bearing, back_left_bearing, back_right_bearing - flat 20 mm wide, 10 deep, on front_edge
top_beam_1_2_3_5_6	2000 x 90 x 45 mm oak, 5 pieces: top_beam_1, top_beam_2, top_beam_3, top_beam_5, top_beam_6
top_beam_4_7	2000 x 90 x 45 mm oak, 2 pieces: top_beam_4, top_beam_7 - 2 x dowel hole ø20, 26 deep, in front_edge, 467.5 and 1532.5 from left_end; centered
tray_front_back	2000 x 90 x 20 mm oak, 2 pieces: tray_front, tray_back - 2 x dowel hole ø10, 21 deep, in front_edge, 467.5 and 1532.5 from left_end; centered
tray_left_right	245 x 90 x 20 mm oak, 2 pieces: tray_left, tray_right
tray_bottom	1960 x 245 x 20 mm oak
top_beam_dowel	ø20 x 50 mm oak, 4 pieces: top_beam_4_dowel_left, top_beam_7_dowel_left, top_beam_4_dowel_right, top_beam_7_dowel_right
tray_front_dowel	ø10 x 40 mm oak, 2 pieces: tray_front_dowel_left, tray_front_dowel_right

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.workbench
```

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

```
def workbench(
    length: float = 2000,
    depth: float = 600,
    height: float = 900,
    overhang: float = 400,
    stretcher_width: float = 300,
    stretcher_height: float | None = None,
    lower_rail_height: float = 120,
    leg_cap: float = 30,
    tusks: int = 2,
    top_beams: int = 7,
    tray_thickness: float = 20,
) -> Assembly:
    """A knock-down workbench, an approximation of Rex Krueger's Quick Stack
    workbench (get his plans for the real thing, at rexkrueger.com).
```

Two leg assemblies, `length` - 2 x `overhang` apart over the outsides of the legs, are joined by two wide stretchers with tusk tenons, and carry a split top `height` above the floor: a massive desk at the front and a tray (a drawer-like box) at the back, flush with the legs at the front and the back, `depth` deep, `overhang` beyond the legs at each end.

Each leg assembly: two 90 mm square legs, their feet chamfered, and two rails from front to back, all glued up from two 45 mm boards (the legs' glue lines on their sides, the rails' upright), the rails flush with the legs' front and back, lapped across the legs but let in only partly: the rails are cut to half their thickness, the legs notched a quarter of theirs, so the rails stand out of the legs by a quarter. The lower rail, as square as the legs, is on the outside, `lower\_rail\_height` above the floor. The upper one, 90 x 135 mm, on the inside and flush

with the tops of the legs, has its laps cut to half dovetails, flat at the bottom and sloped at the top: the laps are cut back `leg\_cap` from the top, so that the legs keep a solid cap above the notches.

The stretchers, 45 mm thick and `stretcher\_width` wide, stand on the lower rails (or `stretcher\_height` above the floor). Their long through tenons, as thick as the stretchers and a third as wide, stand out of the legs by as much as the legs are wide, their corners chamfered big. Each tenon is held by `tusks` dowel tusks (in the second and the fourth fifth of the tenon), level through its thickness, their heads to the front: 20 mm dowels a bit longer than the legs are wide, their flats tapering over 50 mm at their tips, bearing on a half dowel glued flat side down on the leg behind the tenon.

The desk is glued up from `top\_beams` 45 x 90 mm beams standing on edge (as "2x4s"). The tray, of `tray\_thickness` boards, has walls as high as the desk is thick and a bottom between them, flush with their lower edges, so that it can be turned over: open side up it holds tools, upside down it is a flat part of the top. Both sit on the legs and the upper rails, located by short dowels in the rails: two 20 mm ones into the desk (into two of its beams) and a 10 mm one into the tray's front wall on each rail; the back wall has matching holes in its top edge for the tray turned over.

"""

```
leg_size, beam = BENCH_LEG_STOCK.thickness, BENCH_STOCK.thickness
beam_height = 2 * beam
leg_length = height - beam_height
span = length - 2 * overhang # over the outsides of the legs
if stretcher_height is None:
    stretcher_height = lower_rail_height + leg_size
root = orient(at=(overhang, 0, 0), right_end=UP, top_face=LEFT)
asm = Assembly("workbench", root_location=root)
```

```
# The leg assemblies; the right one is the mirror image of the left one.
legs: dict[str, dict[str, Board]] = {}
upper_rails: dict[str, Board] = {}
lap = dict(lap_thickness=leg_size / 2, over=True, proud=leg_size / 4)
for side, mirror in [("left", False), ("right", True)]:
    front, back = (
        asm.add(
            Board(
```

```

        f"{side}_{end}_leg",
        BENCH_LEG_STOCK,
        leg_length,
        leg_size,
        mirror=mirror,
        plies=2,
        plies_across="width",
    )
)
for end in ("front", "back")
)
legs[side] = {"front": front, "back": back}
for leg in (front, back):
    leg.cut(EndChamfer(leg.left_end, size=5))
between = depth - 2 * leg_size
lower, upper = (
    asm.add(
        Board(
            f"{side}_{level}_rail", BENCH_LEG_STOCK, between, width, mirror=mirror, plies=2
        )
    )
    for level, width in [("lower", leg_size), ("upper", 1.5 * leg_size)]
)
upper_rails[side] = upper
for end, leg in [(lower.left_end, front), (lower.right_end, back)]:
    asm.join(
        HalfLap(
            lap=end,
            socket=leg.back_edge if leg is front else leg.front_edge,
            position=offset(lower_rail_height, ref=leg.left_end),
            flush=(leg.top_face, lower.top_face),
            **lap,
        )
    )
)
for end, leg in [(upper.left_end, front), (upper.right_end, back)]:
    asm.join(
        DovetailHalfLap(
            lap=end,
            socket=leg.back_edge if leg is front else leg.front_edge,
            position=flush_with(leg.right_end),
            flush=(leg.bottom_face, upper.top_face),

```

```

        stopped={leg.right_end: leg_cap},
        slope=6,
        sloped=(upper.back_edge,),
        **lap,
    )
)

# The stretchers, their reference faces outside, through the inner faces of the legs.
tenon_width = stretcher_width / 3
for end in ("front", "back"):
    stretcher = asm.add(
        Board(f"{end}_stretcher", BENCH_STOCK, span - 2 * leg_size, stretcher_width)
    )
    for side, tenon in [("left", stretcher.left_end), ("right", stretcher.right_end)]:
        leg = legs[side][end]
        name = f"{end}_{side}"
        tusk_rods = tuple(
            asm.add(Rod(f"{name}_tusk_{i + 1}", BENCH_DOWEL_STOCK, length=110))
            for i in range(tusks)
        )
        bearing = asm.add(Rod(f"{name}_bearing", BENCH_DOWEL_STOCK, length=tenon_width))
        asm.join(
            DowelTuskTenon(
                tenon=tenon,
                mortise=leg.bottom_face,
                position=offset(stretcher_height, ref=leg.left_end),
                flush=(
                    (leg.front_edge, stretcher.top_face)
                    if end == "front"
                    else (leg.back_edge, stretcher.top_face)
                ),
                across=CENTERED,
                shoulder=(stretcher_width - tenon_width) / 2,
                tenon_thickness=beam,
                tenon_chamfer=3,
                tenon_corner_chamfer=25,
                tusks=tusk_rods,
                bearings=(bearing,),
                tusk_head=leg.front_edge,
                tusk_slope=12.5,
                flat_length=50,
            )
        )

```

```

        protrusion=leg_size,
    )
)

# The desk: beams on edge, their reference faces to the front, glued face to face.
beams = [
    asm.add(Board(f"top_beam_{i + 1}", BENCH_STOCK, length, beam_height))
    for i in range(top_beams)
]
for a, b in itertools.pairwise(beams):
    asm.join(Butt(a.bottom_face, b.top_face))

# The tray: the front and the back as long as the bench, the ends between
# them, the bottom between all four, flush with their lower edges; the
# reference faces outside.
tray_stock = Stock.board(thickness=tray_thickness, species="oak")
tray_front_y = top_beams * beam
tray_depth = depth - tray_front_y
inside = tray_depth - 2 * tray_thickness
tray_front, tray_back = (
    asm.add(Board(f"tray_{end}", tray_stock, length, beam_height)) for end in ("front", "back")
)
tray_ends = []
for end in ("left", "right"):
    wall = asm.add(Board(f"tray_{end}", tray_stock, inside, beam_height))
    tray_ends.append(wall)
    asm.join(
        Butt(
            tray_front.bottom_face,
            wall.left_end,
            position=flush_with(getattr(tray_front, f"{end}_end")),
            flush=(tray_front.front_edge, wall.front_edge),
        )
    )
)
for wall, outer in zip(
    tray_ends, (tray_ends[0].bottom_face, tray_ends[1].top_face), strict=True
):
    asm.join(
        Butt(
            wall.right_end,
            tray_back.bottom_face,

```

```

        position=flush_with(outer),
        flush=(wall.front_edge, tray_back.front_edge),
    )
)
bottom = asm.add(Board("tray_bottom", tray_stock, length - 2 * tray_thickness, inside))
asm.join(
    Butt(
        tray_front.bottom_face,
        bottom.front_edge,
        flush=(tray_front.front_edge, bottom.bottom_face),
    )
)

# Dowels into the tops of the upper rails, which start behind the front
# legs (at y = leg_size), in line with the legs but for the quarter
# standing out of them.
for side in ("left", "right"):
    rail = upper_rails[side]
    inner = overhang + leg_size if side == "left" else length - overhang - leg_size
    x = inner - 3 * leg_size / 4 if side == "left" else inner - leg_size / 4
    # A thinner dowel into the tray's wall, a third of its thickness.
    for board, front_y, stock, dowel_length in [
        (beams[top_beams // 2], top_beams // 2 * beam, BENCH_DOWEL_STOCK, 50),
        (beams[-1], (top_beams - 1) * beam, BENCH_DOWEL_STOCK, 50),
        (tray_front, tray_front_y, TRAY_DOWEL_STOCK, 40),
    ]:
        dowel = asm.add(Rod(f"{board.name}_dowel_{side}", stock, length=dowel_length))
        asm.join(
            Dowels(
                board.front_edge,
                rail.back_edge,
                dowels=(dowel,),
                offsets=(front_y + board.thickness / 2 - leg_size,),
                flush=(board.top_face, rail.left_end),
                across=offset(leg_size - front_y),
                position=offset(x, ref=board.left_end),
            )
        )
    )

# The tray's back wall comes over the rail when the tray is turned over.
tray_back.cut(
    Hole(

```

```
        tray_back.back_edge,  
        TRAY_DOWEL_STOCK.diameter,  
        depth=21,  
        at={tray_back.left_end: length - x - leg_size / 2},  
        kind="dowel hole",  
        note="for the tray turned over",  
    )  
)  
return asm
```