

Carpentry Compiler: Supplemental Material

CHENMING WU, Tsinghua University and University of Washington

HAISEN ZHAO, University of Washington

CHANDRAKANA NANDI, University of Washington

JEFFREY I. LIPTON, University of Washington

ZACHARY TATLOCK, University of Washington

ADRIANA SCHULZ, University of Washington

ACM Reference Format:

Chenming Wu, Haisen Zhao, Chandrakana Nandi, Jeffrey I. Lipton, Zachary Tatlock, and Adriana Schulz. 2019. Carpentry Compiler: Supplemental Material. *ACM Trans. Graph.* 38, 6, Article 195 (November 2019), 16 pages. <https://doi.org/10.1145/3355089.3356518>

1 OPTIMIZATION RESULTS

In this section, we provide additional details and results of our optimization method for generating fabrication instructions for all the models in Figure 9.

1.1 Cost Metric Parameters

The material cost is measured as the sum of the costs of all used lumber pieces, $l_i, i = 1, \dots, N$. The price of each lumber, $P(l_i)$ depends on its dimension and is given in Table S1. The prices are relative and are computed based on prices from standard US vendors.

$$f_c = \sum_{i=1}^n P(l_i).$$

To compute f_t , we assign fabrication times to each tool based on the complexity of the setup and operating processes as reported by our carpentry experts. Chopsaw has the simplest setup process and therefore the least setup time. It is followed by bandsaw and jigsaw have a similar setup process where the path has to be precisely marked on the part. Both are more time consuming than setting up a chopsaw. Tracksaw has the most difficult setup and is thus assigned the largest time. Chopsaw and tracksaw allow partial setup where only some parameters are changed. In these cases, the time is half the full setup time. The operating times are estimated to be weights of the setup times. The number of fabrication times are reported in Table S2. We have relative fabrication time f_t as follows.

Authors' addresses: Chenming Wu, wcm15@mails.tsinghua.edu.cn, Tsinghua University and University of Washington; Haisen Zhao, haisen@cs.washington.edu, University of Washington; Chandrakana Nandi, cnandi@cs.washington.edu, University of Washington; Jeffrey I. Lipton, jilipiton@uw.edu, University of Washington; Zachary Tatlock, ztatlock@cs.washington.edu, University of Washington; Adriana Schulz, adriana@cs.washington.edu, University of Washington.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2019 Association for Computing Machinery.

0730-0301/2019/11-ART195 \$15.00

<https://doi.org/10.1145/3355089.3356518>

Stock	Dimension	Relative Material Cost
2" × 4"	24"	0.30
2" × 4"	48"	0.55
2" × 4"	96"	1.00
2" × 2"	24"	0.30
2" × 2"	48"	0.55
2" × 2"	96"	1.00
4" × 4"	24"	0.75
4" × 4"	48"	1.375
4" × 4"	96"	2.50
2" × 8"	24"	0.75
2" × 8"	48"	1.375
2" × 8"	96"	2.50
1/2"	12" × 20"	0.55
1/2"	24" × 20"	1.0
1/2"	48" × 96"	6.5
3/4 "	12" × 20"	0.7
3/4 "	24" × 20"	1.2
3/4 "	48" × 96"	8.5

Table S1. Prices of stocks

Tool	Full Fab. Time	Partial Fab. Time
Chopsaw	1.0	0.5
Bandsaw	4.0	N/A
Jigsaw	10.0	N/A
Tracksaw	5.0	2.5
Drill	2.0	N/A

Table S2. Relative fabrication time costs for different tools.

$$f_t = \sum_k n_k \cdot c_k$$

where k is the tool type, for example, n_c and n_{cp} are the number of the full setups and partial setups for chopsaw respectively; c_c and c_{cp} are the corresponding costs of full setups and partial setups for chopsaw respectively.

To compute f_p , we rely on the precision levels from Table S4 to assign a quantitative value for error-per-cut to each tool as shown in Table S3. Lower values are better (we only consider errors in measurement, not errors in fabrication, or stochastic errors).

The minimum measurement that can be made accurately using our tools is $m = 1/16''$ for all dimensions, and $m = 1$ for all angles

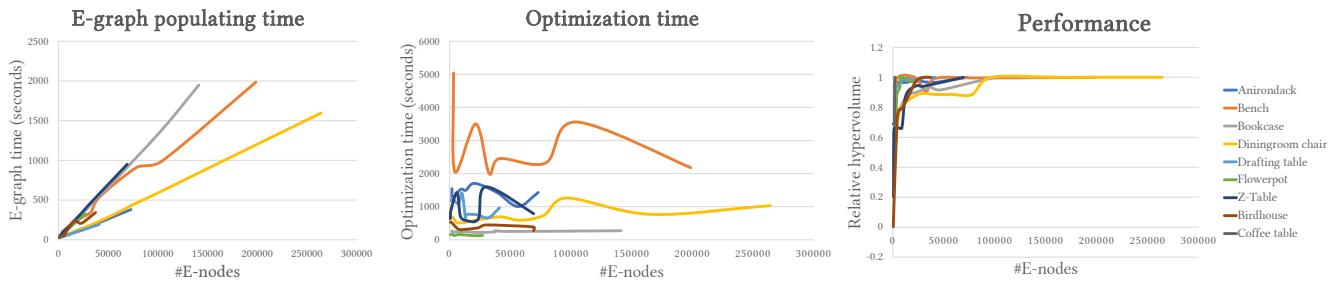


Fig. S1. From left to right: 1) time versus #e-nodes for populating e-graphs; 2) optimization time versus #e-nodes; 3) relative hypervolume versus #e-nodes.

Tool	Relative Error per cut
Tracksaw	1.0
Chopsaw	1.0
Bandsaw	2.5
Jigsaw	6.25

Table S3. Relative error per cut for each tool. Lower values indicate less error per cut.

(using tape measure and protractor). Thus, measurements that are a multiple of m have zero error. The error, ϵ for all other measurements, m' is computed by the following formula: $\epsilon = m' \% m$. The mean precision is then measured as:

$$f_p = \frac{\sum_{i=1}^n (1 + \epsilon_i) \cdot p_i}{n}$$

where n is the number of cuts and p_i is the precision of the tool used for the i^{th} cut.

1.2 Results and Comparison to Expert Fabrication Plan

The additional results for the models in Figure 9 are shown in Figures S2 and S3. These results show some additional insights into the performance of the algorithm. The drafting table (9.B) example illustrates the trade-off between fabrication time and material cost. In solution B, by using smaller stocks, the parts on the block can be arranged in a way that reduces the setups, which reduces fabrication time. Solution A, on the other hand, uses fewer and larger boards and therefore, the material cost is reduced at the expense of fabrication time. For the birdhouse (9.D), HELM's result reuses many of the same setups in solution B, saving time, while HELM improved precision by optimizing the order of cuts most influenced by kerf in solution A.

1.3 Optimization Time and Convergence

Cut planning is a combinatorial problem. The use of e-graphs and our pruning strategy make the problem tractable, but as discussed in the limitations section, this may not always return the optimal fabrication plan. To further evaluate our method, we vary the amount of pruning and plot graphs that show 1) performance versus the number of e-nodes (using the hypervolume indicator for multi-objective optimization [Auger et al. 2009]) and 2) computation time

versus the number of e-nodes. As hypervolume varies from different designs due to different choices on reference points, we propose a normalization method to visualize the results - for each design d_i , we concatenate the scores of all Pareto-fronts, and use the concatenated matrix to compute its reference point ref_i . A maximal hypervolume among all Pareto-fronts calculated by ref_i is hv'_i , which can be used to normalize all other hypervolumes into a range of $[0, 1]$.

The computation time for computing the e-graph grows linearly with the number of e-nodes, while the performance quickly increases and then tapers off (see the right-most sub-figure in Figure S1). It is worth noting that the number of programs that an e-graph can represent is much greater than the number of e-nodes. We also reported time for populating e-graphs and optimization in Figure S1.

1.4 Concurrent Optimization of Multiple Models

To evaluate the convergence of our technique, we propose optimizing the fabrication process of multiple models. As an example, we consider the fabrication process of a dining set, consisting of a table (Figure 9.H) and six chairs (Figure 9.F). We argue that it is increasingly challenging for experts to optimize the fabrication instructions as the number of parts increases and, therefore, do not compare to expert results in this example. Instead, we compare running our algorithms for fabricating all parts in the dining set to the result of optimizing each object (chairs and table) individually. For the second case, we take the resulting Pareto-fronts and add them by considering all possible combinations of Pareto-optimal plans for each object.

While we expect that the first case (concurrent optimization) should generate better results since the search is done on a larger space, this may not be the case because of the need for additional punning. The number of parts in the dining set is 108, which took 1.4 hours to populate e-graphs and additional 10.1 hours to optimize (#e-nodes is 359,489). The running time of optimization is high because the height of ASTs depends on the number of parts, and a tall AST needs more time to do cross-overs and mutations in our algorithm. Comparatively, we try all possible combinations of the points on Pareto-fronts from Section 2.2 (#e-nodes are 60,272 and 1,241) to construct a new Pareto-front for fabricating the whole dining set. Although concurrent optimization takes a substantial amount of time, since it gives results that fully dominate the ones from added

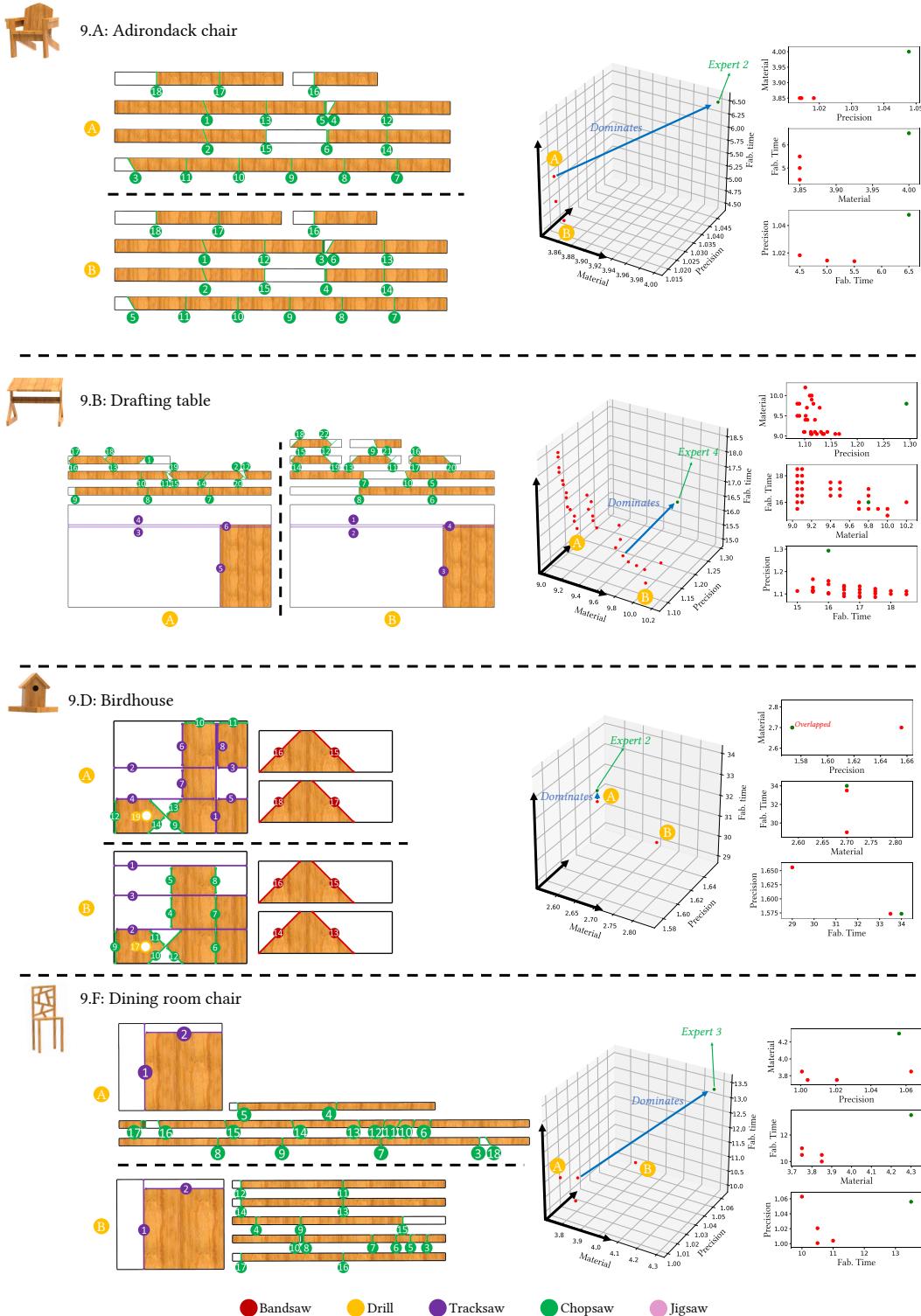


Fig. S2. Additional optimization results.

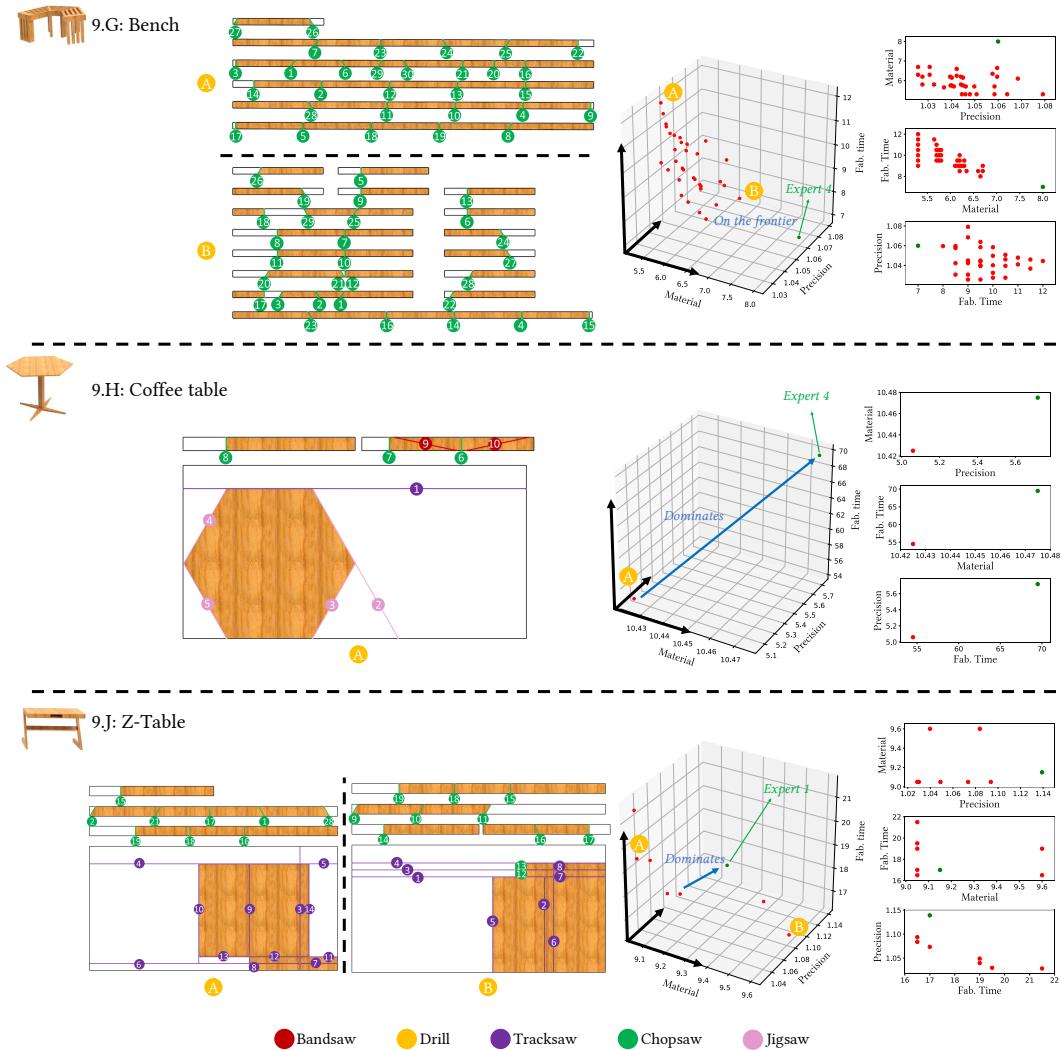


Fig. S3. Additional optimization results (continued).

Pareto-front exploration, it still yields significant benefits because relative to actual fabrication time, the optimization time is negligible. A smaller number of dining set (57 parts, Figure S4: bottom) was also tested using the described approach, and a similar result is obtained.

As an additional experiment, we also took the points from the separately-optimized pieces of the dining set, concatenated them into a single fabrication plan, and then further optimized their cut ordering to reduce fabrication time. This alternate strategy actually uncovered the lowest cost solutions with respect to fabrication time, demonstrating that the concurrent optimization strategy is not always able to explore the full Pareto frontier for large search spaces. We are eager to further explore how different optimizations scale in future work. We speculate that the difference between the concurrent strategy (red dots in Figure S4) and this strategy (blue dots in Figure S4) is that pruning is done by optimizing packing which reduces material costs, but makes it more challenging to find solutions that trade-off more material cost for fabrication time once the search space grows too large.

1.5 Implementation Details

To make our paper self-contained, we will elaborate implementation details from several perspectives in this section.

This paragraph gives more details of “Packing Pieces onto Stock”. Given a library of stock lumbers, we first group them by their dimensions. The parts are assigned to these groups using the method of Section 5.2. One part might be assigned to different groups, and we only choose the group with the maximum space utilization for this part. With the packing process of Section 5.2, our algorithm packs all parts to the first stock lumber (maximal bounding volume lumber) of the current group. The packing process should continue until 1) it is impossible to place more parts or 2) all parts are placed. Once the process stops, a resulting packing is received which we call an *Arrangement*. If there are still un-placed parts, our algorithm uses another void first stock lumber to place the following parts. This packing process will not terminate unless 1) all parts are placed into specific stock lumber or 2) the following parts cannot be placed into void stock lumber.

Such a packing process traverses all of the parts in a specific sequence, which we call a full *Traversal*. At the same time, we get a set of *Arrangements*. If these *Arrangements* contain all parts of the current group, we repeat the packing process to pack the parts of each *Arrangement* into smaller stocks of the lumber group. With such a recursive packing process, e-graphs can be constructed by organizing all of the obtained *Arrangements* as stock e-nodes.

The key step of the packing process is to select the next part and its orientation to place into the current stock lumber. There are many possible choices for selecting the next part and its orientation. Our algorithm packs every un-placed part in each of its suitable orientations, then sorts all of the valid *Arrangements* following a radix sorting manner, which first sorts them by the number of aligned edges and then sort them by their bounding volume and distance to stock boundaries for those *Arrangements* with the same number of aligned edges. With different choices of the next part and its orientation, different *Arrangements* can be obtained. So the whole packing process is indeed a tree search process. Suppose that the

number of input parts is N , and each part has M valid orientations to assign to the current stock lumber, which results in $N! \times M^N$ search space. It is time-consuming to enumerate all of these possible packing results with such a large search space. We use the number of *Traversal T* as our termination criterion. We use the same T as 50 in our experiments of Figure S2 and S3. The number of e-classes will increase with a greater T . As explained in Section 5.2, we use a heuristic to prune some of the results according to the number of aligned edges. For each e-class, we keep the top n stock e-nodes ($n = 10$ is adopted in our experiments). For each stock e-nodes, we only compile it down for maximal P different orders of cuts, and we found $P = 25$ could give satisfactory results.

Our system adopts a multi-objective genetic algorithm, which could be easily parallelized. In our implementation, we create multiple threads to run the NSGA-II based optimization. We set the parameters of our genetic algorithm as $p_c = 0.95$, $p_m = 0.1$. Each thread maintains a population which has 120 active individuals. Our genetic algorithm has two termination criteria: 1) no performance gain within L iterations, or 2) exceed K iterations, whichever is first satisfied. In our experiments, we set $L = 800$ and $K = 16,000$. Moreover, a fast convergence sometimes requires proper strategies of initializing populations. In practice, carpenters would try to use a minimal set of tools to fabricate a design. Borrowing this idea, we use a portion of threads to initialize population randomly, but try to use a specific process as many as possible, while the other threads completely randomize initialization similar to what a standard genetic algorithm does.

2 LANGUAGE DETAILS AND EXAMPLES

This section expands on Section 4 by defining each operation and parameter of the languages shown in Figure 4 in the paper, describes process characterization and the mapping from HL-HELM to LL-HELM.

2.1 High-Level HELM

Figure 4 (left) shows the grammar for HL-HELM programs, that are composed of a sequence of assignments that bind identifiers to the result of high-level fabrication operations: *Make_Stock*, *Make_Hole*, and *Make_Cut*. Our language is inspired by standard feature-based CAD scripting languages [FeatureScript 2019], where the features map to fabrication operations, such as getting stock and making cuts. As in CAD languages, we also include a special statement for defining 2D sketches. CAD systems create 3D geometry by applying features (e.g., extrusion or drafting) to 2D sketches. *Make_Hole* and *Make_Cut* use *sketches* to specify the path an operation should follow, e.g., a polyline to specify a bandsaw cut. Sketches are defined by a set of geometric primitives (points, line segments, circles, splines) and constraints.

A sketch is defined on the face on which an operation occurs. This information is provided by a *query*. A query is always made on a specific body. Every operation (*Make_Hole*, *Make_Cut*) takes an id for a body and all operations implicitly refers to that body. Some examples of queries are shown in Figure 4 (left).

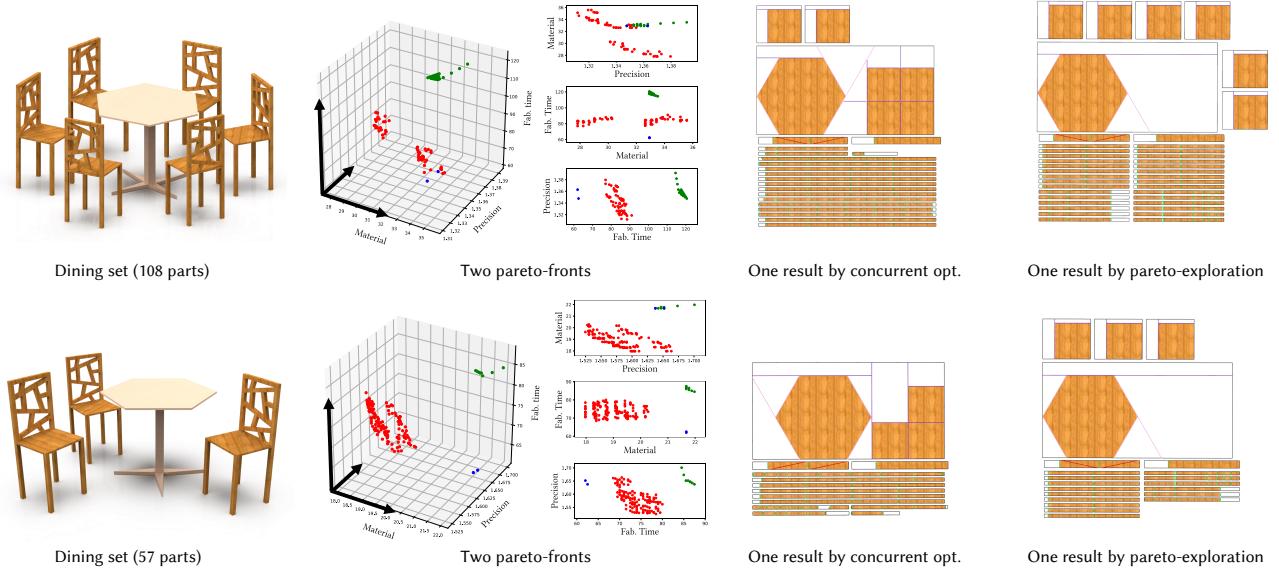


Fig. S4. From left to right: dining set example including six chairs and one table; Two corresponding pareto-fronts, red dots correspond to the results of concurrent optimization, and green dots correspond to the results of pareto-fronts exploration. Note that the showing results on the right-hand side are the ones with minimal fabrication time.

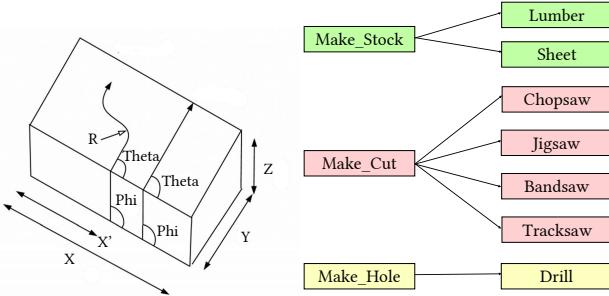


Fig. S5. (a) Process characterization diagram for saws. (b) Surjective mapping from HL-HELM to LL-HELM in our compiler.

2.2 Low-Level HELM

It supports fabrication operations like *Chopsaw*, *Jigsaw*, *Bandsaw*, and *Drill* (hand drill). Features in HL-HELM like *Make_Cut* are mapped to concrete LL-HELM operations like *Chopsaw* and *Jigsaw*. Similarly, *Make_Hole* is mapped to *Drill*, and *Make_Stock* is mapped to *Lumber* or *Sheet*, depending on the dimensions. Lumber and sheet are referenced using a *catalog_id* that uniquely identifies the part from a library of available material.

Some of these fabrication operations require a setup that configures the tool for performing the tasks. For example, the setup for chopsaw is specified by two angles: the *miter angle*, the *bevel angle*, and the offset from the stop block; and the setup for a drill is specified by the diameter of the hole. The diameter determines the choice of the drill bit. The bandsaw and jigsaw in Figure 4 do not

require a setup¹. The cutting path for these two tools is specified by a list of reference points (*ref_pt*), where a *ref_pt* is defined by the intersection of two lines. These lines are specified by an offset from an edge on the part. Drill also requires a *ref_pt* that specifies the point where the hole should be made.

Additionally, LL-HELM supports a *Stack* operation that places two parts together before performing an operation when both parts use the same setup. This helps reduce the number of times the setup needs to be changed which is useful because changing setup can be time-consuming and error-prone. The language also provides an *Unstack* operation that separates the stacked parts after an operation has been performed. A LL-HELM program is then, a sequence of setups and assignments, followed by a *Return* statement that returns the resulting parts obtained from the fabrication operations.

2.3 Process Characterization

Table S4 describes the process characterization for every operation in our pipeline. The first four rows summarize the saw operations. Tracksaw and chopsaw are the most precise of all the cutting operations we support, followed by bandsaw and jigsaw. *x* and *y* are the maximum lengths the tool can support the respective dimension. *z* is the maximum height of the part that can be fit under the tool. *x'* is the maximum distance between the leftmost end of the tool's platform and the part. In the case of jigsaws, the maximum *x* and *y* dimensions are ∞ since the tool does not in any way constrain the part along *x* or *y*. Due to the same reason, *x'* is also not constrained. The values for the other tools are in the table. The value of *z* is 1 inch for jigsaw and tracksaw. For chopsaw, it is 4 inches and for bandsaw it is 6 inches. Theta represents the miter angle and phi

¹Note that some bandsaws may allow setup where the table can be rotated by an angle.

Tool	Precision	x	y	z	x'	Theta	Phi	R	Partial	Curve	Internal
Jigsaw	low	(0, ∞)	(0, ∞)	(0, 1 '')	(0, ∞)	(0, 180)	90	(1 '', ∞)	T	T	T
Chopsaw	high	(0, 96 '')	(0, 6 '')	(0, 4 '')	(0, 36 '')	(-50, 60)	(45, 135)	-	F	F	F
Bandsaw	medium	(0, 26 '')	(0, 24 '')	(0, 6 '')	(0, 13 '')	(0, 180)	90	(1 '', ∞)	T	T	F
Tracksaw	high	(0, 96 '')	(0, 48 '')	(0, 1 '')	(0, 36 '')	90	(45, 135)	-	F	F	F
Drill	high	(0, ∞)	(0, ∞)	(0, ∞)	(0, ∞)	-	-	drill-bit diameter	-	-	-

Table S4. Process characterization table for all tools in LL-HELM.

represents the bevel angle. R is the minimum curvature of a path that the tool can follow. For chopsaw and tracksaw curved cuts are not possible, but for bandsaw and jigsaw the values are shown in the table. Figure S5(a) illustrates these parameters. Both jigsaw and bandsaw support partial cuts and curves whereas chopsaw and tracksaw do not. “Internal” indicates whether the tool can be used to make an internal cut on a part. Only jigsaws can be used to perform such cuts.

2.4 Mapping from HL-HELM to LL-HELM

There is an explicit surjective mapping from every feature in HL-HELM to an operation in LL-HELM. This mapping is shown in Figure S5(b). There are three types of features in our current implementation of HL-HELM which is easily extensible as Section 4 explains. In the figure, green denotes stock allocation, red denotes cuts, and yellow denotes holes.

2.5 LL-HELM Interpretation

LL-HELM instructions use UIDs to represent geometric information such as edges and faces, which are not difficult to interpret by carpenters. We build a LL-HELM visualization UI to depict step-by-step instructions that carpenters will follow. The left side of our UI shows optimized LL-HELM codes, and the right side visualizes a specific line of codes that needs to be performed. Users can easily explore the whole programs by clicking “previous step” and “next step” buttons.

2.6 HL- and LL-HELM of Designs

We show the high-level HELM for all of the models in Figure 9 and the resulting low-level HELM programs in Listings 1-20, which are at the end of the supplemental material. The principle of selecting a representative LL-HELM is choosing the one with the lowest cost on fabrication time from pareto-fronts.

2.7 User study

To evaluate the expressiveness of our system, we asked the three experts who created the physical models shown in the teaser, to fill out a survey on their experience with the tool and how it compares to conventional CAD systems. Here is a snippet from one of the experts:

I find it much easier to produce arbitrary models in CAD. However, I found the production of models with this tool to be much easier than CAD if I would later be constructing them from lumber. It is very easy to make a sphere in most CAD programs, and it is very difficult to construct one in this tool. Likewise, in the real world, producing a perfect sphere out of a 2x4 is nontrivial. When drawing the Adirondack chair,

I saved significant time because the tool contained predefined stock lumber sizes. I could simply load a bunch of stock into the environment, cut it, and stitch it together in the assembly environment. Drawing the chair in CAD would be more difficult as I would have to specify additionally dimensions for each piece.”

REFERENCES

- Anne Auger, Johannes Bader, Dimo Brockhoff, and Eckart Zitzler. 2009. Theory of the Hypervolume Indicator. In *Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms (FOGA ’09)*. ACM, New York, NY, USA, 87–102. <https://doi.org/10.1145/1527125.1527138>
- FeatureScript. 2019. Welcome to FeatureScript. (2019). <https://cad.onshape.com/FsDoc/>.

HL-HELM AND LL-HELM PROGRAMS

Listing 1. Figure 9.A - Adirondack chair (HL-HELM)

```

1 Box005 = Make_Stock(457.2, 38.1, 88.9);
2 MyLine000 = Line(-22.7087, -38.1, 36.576, 0);
3 Sketch010 = Make_Sketch(
4     Query_Face_By_Closest_Point(Box005, 228.6, 19.05, 0),
5     Geometry(MyLine000),
6     Constraint(PointOnObject(Start(MyLine000),
7         Query_Edge_By_Closest_Point(Box005, 228.6, 38.1, 0)),
8         PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
9             Box005, 228.6, 0, 0)), Angle(End(MyLine000), Start(
10            Query_Edge_By_Closest_Point(Box005, 228.6, 0, 0)), 110),
11            Distance(End(MyLine000), End(Query_Edge_By_Closest_Point(
12            Box005, 228.6, 0, 0)), 420.624)));
13 Cut010 = Make_Cut(Box005, Sketch010, 1);
14 Box004 = Make_Stock(457.2, 38.1, 88.9);
15 MyLine001 = Line(434.491, -38.1, 420.624, 0);
16 Sketch009 = Make_Sketch(
17     Query_Face_By_Closest_Point(Box004, 228.6, 19.05, 0),
18     Geometry(MyLine001),
19     Constraint(PointOnObject(Start(MyLine001),
20         Query_Edge_By_Closest_Point(Box004, 228.6, 38.1, 0)),
21         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
22            Box004, 228.6, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
23            Box004, 228.6, 0, 0)), End(MyLine001), 110), Distance(End(
24            MyLine001), Start(Horizontal), 420.624));
25 Cut009 = Make_Cut(Box004, Sketch009, 0);
26 Box020 = Make_Stock(457.2, 38.1, 88.9);
27 MyLine002 = Line(34.0176, 0, 85.344, 88.9);
28 Sketch008 = Make_Sketch(
29     Query_Face_By_Closest_Point(Box020, 228.6, 0, 44.45),
30     Geometry(MyLine002),
31     Constraint(PointOnObject(Start(MyLine002),
32         Query_Edge_By_Closest_Point(Box020, 228.6, 0, 0)),
33         PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
34            Box020, 228.6, 0, 88.9)), Angle(Start(
35            Query_Edge_By_Closest_Point(Box020, 228.6, 0, 0)), Start(
36            MyLine002), 60), Distance(End(MyLine002), End(
37            Query_Edge_By_Closest_Point(Box020, 228.6, 0, 88.9),
38            371.856)));
39 Cut008 = Make_Cut(Box020, Sketch008, 1);
40 Box021 = Make_Stock(457.2, 38.1, 88.9);
41 MyLine003 = Line(0, 0, 32.357, 88.9);
42 Sketch007 = Make_Sketch(
43     Query_Face_By_Closest_Point(Box021, 228.6, 0, 44.45),
44     Geometry(MyLine003),
45     Constraint(Coincident(Start(MyLine003), Start(Horizontal)),
46         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
47            Box021, 228.6, 0, 88.9)), Angle(Start(
48            Query_Edge_By_Closest_Point(Box021, 0, 0, 44.45)), 20));
49 Cut007 = Make_Cut(Box021, Sketch007, 1);
50 Box010 = Make_Stock(457.2, 38.1, 88.9);
51 MyLine004 = Line(32.357, 0, 0, 88.9);
52 Sketch006 = Make_Sketch(
53     Query_Face_By_Closest_Point(Box010, 228.6, 0, 44.45),
54     Geometry(MyLine004),
55     Constraint(PointOnObject(Start(MyLine004),
56         Query_Edge_By_Closest_Point(Box010, 228.6, 0, 0)),
57         Coincident(End(MyLine004), End(Query_Edge_By_Closest_Point(
58            Box010, 0, 0, 44.45))), Angle(Start(
59            Query_Edge_By_Closest_Point(Box010, 228.6, 0, 0)), Start(
60            MyLine004), 110));
61 Cut006 = Make_Cut(Box010, Sketch006, 1);
62 Box019 = Make_Stock(457.2, 38.1, 88.9);
63 MyLine005 = Line(85.344, 0, 34.0176, 88.9);
64 Sketch = Make_Sketch(
65     Query_Face_By_Closest_Point(Box019, 228.6, 0, 44.45),
66     Geometry(MyLine005),
67     Constraint(PointOnObject(Start(MyLine005), Query_Edge_By_Closest_Point(
68       Box019, 228.6, 0, 0)), PointOnObject(End(MyLine005),
69       Query_Edge_By_Closest_Point(Box019, 228.6, 0, 88.9)), Angle(
70       Start(Query_Edge_By_Closest_Point(Box019, 228.6, 0, 0)), Start(
71       MyLine005), 120), Distance(Start(MyLine005),
72       Query_Edge_By_Closest_Point(Box019, 457.2, 0, 44.45), 371.856));
73 Cut = Make_Cut(Box019, Sketch, 1);
74 Box009 = Make_Stock(670.56, 38.1, 88.9);
75 MyLine006 = Line(-670.56, 0, -638.203, 88.9);
76 Sketch004 = Make_Sketch(
77     Query_Face_By_Closest_Point(Box009, 335.28, 38.1, 44.45),
78     Geometry(MyLine006),
79     Constraint(Coincident(Start(MyLine006), Start(
80        Query_Edge_By_Closest_Point(Box009, 670.56, 38.1, 44.45))),
81         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
82            Box009, 335.28, 38.1, 88.9)), Angle(Start(
83            MyLine006), Start(MyLine005), 120), Distance(Start(
84            MyLine005), 670.56, 38.1, 44.45), 20));
85 Cut004 = Make_Cut(Box009, Sketch004, 0);
86 Box012 = Make_Stock(381, 38.1, 88.9);
87 Box014 = Make_Stock(381, 38.1, 88.9);
88 Box017 = Make_Stock(457.2, 38.1, 88.9);
89 Box013 = Make_Stock(381, 38.1, 88.9);
90 Box = Make_Stock(381, 38.1, 88.9);
91 Box001 = Make_Stock(457.2, 38.1, 88.9);
92 Box007 = Make_Stock(457.2, 38.1, 88.9);
93 Box002 = Make_Stock(381, 38.1, 88.9);
94 Box003 = Make_Stock(457.2, 38.1, 88.9);

```

```

60 Box022 = Make_Stock(670.56, 38.1, 88.9);
61 MyLine007 = Line(-670.56, 0, -638.203, 88.9);
62 Sketch005 = Make_Sketch(
63     Query_Face_By_Closest_Point(Box022, 335.28, 38.1, 44.45),
64     Geometry(MyLine007),
65     Constraint(Coincident(Start(MyLine007), Start(
66        Query_Edge_By_Closest_Point(Box022, 670.56, 38.1, 44.45))),
67         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
68            Box022, 335.28, 38.1, 88.9)), Angle(Start(MyLine007), Start(
69        Query_Edge_By_Closest_Point(Box022, 670.56, 38.1, 44.45)),
70        20));
71 Cut005 = Make_Cut(Box022, Sketch005, 0);

```

Listing 2. Figure 9.A - Adirondack chair (LL-HELM)

```

1 Setup_Chopaw(0.0000, 20.0000, 25.1261)
2 (a0,a1) = Chopaw(lumber_2x4x96, face_0, edge_0)
3 (a3,a4) = Chopaw(lumber_2x4x96, face_1, edge_1)
4 Setup_Chopaw(0.0000, 20.0000, 35.3769)
5 (a5,a6) = Chopaw(a1, face_2, edge_2)
6 (a7,a8) = Chopaw(a4, face_3, edge_3)
7 Setup_Chopaw(0.0000, 30.0000, 3.5699)
8 (a9) = Chopaw(lumber_2x4x96, face_4, edge_4)
9 Setup_Chopaw(0.0000, 30.0000, 0.9928)
10 (a10) = Chopaw(a6, face_5, edge_5)
11 Setup_Chopaw(0.0000, 0.0000, 15.0000)
12 (a11,a12) = Chopaw(a9, face_6, edge_6)
13 (a13,a14) = Chopaw(a11, face_7, edge_7)
14 (a15,a16) = Chopaw(a13, face_8, edge_8)
15 (a17,a18) = Chopaw(a15, face_9, edge_9)
16 (a19,a20) = Chopaw(a17, face_10, edge_10)
17 Setup_Chopaw(0.0000, 0.0000, 18.0000)
18 (a21,a22) = Chopaw(a5, face_11, edge_11)
19 (a23,a24) = Chopaw(a10, face_12, edge_12)
20 (a25,a26) = Chopaw(a8, face_13, edge_13)
21 (a27) = Chopaw(a7, face_14, edge_14)
22 (a28) = Chopaw(lumber_2x4x24, face_15, edge_15)
23 (a30,a31) = Chopaw(lumber_2x4x48, face_16, edge_16)
24 (a33) = Chopaw(a30, face_17, edge_17)
25 Return(a0,a3,a12,a14,a16,a18,a19,a20,a21,a22,a23,a24,a25,a26,a27,a28,a31,a33)

```

Listing 3. Figure 9.B - Drafting table (HL-HELM)

```

1 Box001 = Make_Stock(521.21, 38.1, 88.9);
2 Box013 = Make_Stock(521.21, 38.1, 88.9);
3 MyLine000 = Line(0, 0, 88.9, 88.9);
4 Sketch014 = Make_Sketch(
5     Query_Face_By_Closest_Point(Box013, 260.605, 0, 44.45),
6     Geometry(MyLine000),
7     Constraint(Coincident(Start(MyLine000), Start(Horizontal)),
8         PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
9            Box013, 260.605, 0, 88.9)), Angle(Start(
10           Query_Edge_By_Closest_Point(Box013, 260.605, 0, 0)), Start(
11           MyLine000), 45)));
12 MyLine001 = Line(0, 0, 88.9, 88.9);
13 Sketch = Make_Sketch(
14     Query_Face_By_Closest_Point(Box001, 260.605, 0, 44.45),
15     Geometry(MyLine001),
16     Constraint(Coincident(Start(MyLine001), Start(Horizontal)),
17         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
18            Box001, 260.605, 0, 88.9)), Angle(Start(
19           Query_Edge_By_Closest_Point(Box001, 260.605, 0, 0)), Start(
20           MyLine001), 45)));
21 Cut = Make_Cut(Box001, Sketch, 1);
22 Cut015 = Make_Cut(Box013, Sketch014, 1);
23 MyLine002 = Line(521.21, 0, 432.31, 88.9);
24 Sketch015 = Make_Sketch(
25     Query_Face_By_Closest_Point(Cut015, 282.83, 0, 44.45),
26     Geometry(MyLine002),
27     Constraint(Coincident(Start(MyLine002), End(
28        Query_Edge_By_Closest_Point(Cut015, 260.605, 0, 0))),
29         PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
30            Cut015, 305.055, 0, 88.9)), Angle(Start(MyLine002), End(
31           Query_Edge_By_Closest_Point(Cut015, 260.605, 0, 0)), 45)));
32 MyLine003 = Line(521.21, 0, 432.31, 88.9);
33 Sketch001 = Make_Sketch(
34     Query_Face_By_Closest_Point(Cut, 282.83, 0, 44.45),
35     Geometry(MyLine003),
36     Constraint(Coincident(Start(MyLine003), End(
37        Query_Edge_By_Closest_Point(Cut, 260.605, 0, 0))),
38         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
39            Cut, 305.055, 0, 88.9)), Angle(Start(MyLine003), End(
40           Query_Edge_By_Closest_Point(Cut, 260.605, 0, 0)), 45)));
41 Cut001 = Make_Cut(Cut, Sketch001, 0);
42 Cut016 = Make_Cut(Cut015, Sketch015, 0);
43 MyLine004 = Line(62.8618, 0, 31.4309, 31.4309);
44 Sketch016 = Make_Sketch(
45     Query_Face_By_Closest_Point(Cut016, 260.605, 0, 44.45),
46     Geometry(MyLine004),
47     Constraint(PointOnObject(Start(MyLine004),
48        Query_Edge_By_Closest_Point(Cut016, 260.605, 0, 0))),
49         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
50            Cut016, 44.45, 0, 44.45)), Perpendicular(
51           Query_Edge_By_Closest_Point(Cut016, 44.45, 0, 44.45),
52           MyLine004), Distance(MyLine004, 44.45));
53 Cut014 = Make_Cut(Cut016, Sketch016, 1);
54 Box009 = Make_Stock(323.09, 38.1, 88.9);
55 MyLine005 = Line(0, 88.9, 32.357, 0);

```

```

35 Sketch013 = Make_Sketch(
36     Query_Face_By_Closest_Point(Box009, 161.545, 0, 44.45),
37     Geometry(MyLine005),
38     Constraint(Coincident(Start(MyLine005), End(
39         Query_Edge_By_Closest_Point(Box009, 0, 0, 44.45))),
40         PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
41             Box009, 161.545, 0, 0)), Angle(Start(MyLine005), Start(
42             Query_Edge_By_Closest_Point(Box009, 161.545, 0, 88.9)), 70));
43 MyLine006 = Line(521.21, 0, 432.31, 88.9);
44 Sketch017 = Make_Sketch(
45     Query_Face_By_Closest_Point(Box014, 260.605, 0, 44.45),
46     Geometry(MyLine006),
47     Constraint(Coincident(Start(MyLine006), Start(
48         Query_Edge_By_Closest_Point(Box014, 521.21, 0, 44.45))),
49         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
50             Box014, 260.605, 0, 88.9)), Angle(Start(MyLine006), End(
51             Query_Edge_By_Closest_Point(Box014, 260.605, 0, 0)), 45)));
52 MyLine007 = Line(521.21, 0, 432.31, 88.9);
53 Sketch002 = Make_Sketch(
54     Query_Face_By_Closest_Point(Box002, 260.605, 0, 44.45),
55     Geometry(MyLine007),
56     Constraint(Coincident(Start(MyLine007), Start(
57         Query_Edge_By_Closest_Point(Box002, 521.21, 0, 44.45))),
58         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
59             Box002, 260.605, 0, 88.9)), Angle(Start(MyLine007), End(
60             Query_Edge_By_Closest_Point(Box002, 260.605, 0, 0)), 45)));
61 Cut002 = Make_Cut(Box002, Sketch002, 0);
62 Cut018 = Make_Cut(Box014, Sketch017, 0);
63 MyLine008 = Line(0, 0, 88.9, 88.9);
64 Sketch018 = Make_Sketch(
65     Query_Face_By_Closest_Point(Cut018, 238.38, 0, 44.45),
66     Geometry(MyLine008),
67     Constraint(Coincident(Start(MyLine008), Start(Horizontal)),
68         PointOnObject(End(MyLine008), Query_Edge_By_Closest_Point(
69             Cut018, 216.155, 0, 88.9)), Angle(Start(
70             Query_Edge_By_Closest_Point(Cut018, 260.605, 0, 0)), Start(
71             MyLine008), 45)));
72 MyLine009 = Line(0, 0, 88.9, 88.9);
73 Sketch003 = Make_Sketch(
74     Query_Face_By_Closest_Point(Cut002, 238.38, 0, 44.45),
75     Geometry(MyLine009),
76     Constraint(Coincident(Start(MyLine009), Start(Horizontal)),
77         PointOnObject(End(MyLine009), Query_Edge_By_Closest_Point(
78             Cut002, 216.155, 0, 88.9)), Angle(Start(
79             Query_Edge_By_Closest_Point(Cut002, 260.605, 0, 0)), Start(
80             MyLine009), 45)));
81 Cut003 = Make_Cut(Cut002, Sketch003, 1);
82 Cut019 = Make_Cut(Cut018, Sketch018, 1);
83 MyLine010 = Line(489.779, 31.4309, 458.348, 0);
84 Sketch019 = Make_Sketch(
85     Query_Face_By_Closest_Point(Cut019, 260.605, 0, 44.45),
86     Geometry(MyLine010),
87     Constraint(PointOnObject(Start(MyLine010),
88         Query_Edge_By_Closest_Point(Cut019, 476.76, 0, 44.45)),
89         PointOnObject(End(MyLine010), Query_Edge_By_Closest_Point(
90             Cut019, 260.605, 0, 0)), Perpendicular(
91             Query_Edge_By_Closest_Point(Cut019, 476.76, 0, 44.45),
92             MyLine010), Distance(MyLine010, 44.45)));
93 Box003 = Make_Stock(323.09, 38.1, 88.9);
94 MyLine011 = Line(0, 88.9, -32.357, 0);
95 Sketch012 = Make_Sketch(
96     Query_Face_By_Closest_Point(Box003, 161.545, 38.1, 44.45),
97     Geometry(MyLine011),
98     Constraint(Coincident(Start(MyLine011), End(
99         Query_Edge_By_Closest_Point(Box003, 161.545, 38.1, 44.45))),
100        PointOnObject(End(MyLine011), Query_Edge_By_Closest_Point(
101            Box003, 161.545, 38.1, 0)), Angle(Start(
102            Query_Edge_By_Closest_Point(Box003, 161.545, 38.1, 88.9)),
103            Start(MyLine011), 70)));
104 MyLine012 = Line(62.8618, 0, 31.4309, 31.4309);
105 Sketch005 = Make_Sketch(
106     Query_Face_By_Closest_Point(Cut001, 260.605, 0, 44.45),
107     Geometry(MyLine012),
108     Constraint(PointOnObject(Start(MyLine012),
109         Query_Edge_By_Closest_Point(Cut001, 260.605, 0, 0)),
110         PointOnObject(End(MyLine012), Query_Edge_By_Closest_Point(
111             Cut001, 44.45, 0, 44.45)), Perpendicular(
112             Query_Edge_By_Closest_Point(Cut001, 44.45, 0, 44.45),
113             MyLine012), Distance(MyLine012, 44.45)));
114 MyLine013 = Line(489.779, 31.4309, 458.348, 0);
115 Sketch004 = Make_Sketch(
116     Query_Face_By_Closest_Point(Cut003, 260.605, 0, 44.45),
117     Geometry(MyLine013),
118     Constraint(PointOnObject(Start(MyLine013),
119         Query_Edge_By_Closest_Point(Cut003, 476.76, 0, 44.45)),
120         PointOnObject(End(MyLine013), Query_Edge_By_Closest_Point(
121             Cut003, 260.605, 0, 0)), Perpendicular(
122             Query_Edge_By_Closest_Point(Cut003, 476.76, 0, 44.45),
123             MyLine013), Distance(MyLine013, 44.45)));
124 Cut012 = Make_Cut(Box003, Sketch012, 1);
125 Box011 = Make_Stock(874.78, 38.1, 88.9);
126 Box010 = Make_Stock(609.6, 25.4, 12.7);
127 Box004 = Make_Stock(609.6, 952.5, 12.7);
128 Cut004 = Make_Cut(Cut003, Sketch004, 0);
129 Cut005 = Make_Cut(Cut001, Sketch005, 1);
130 Box = Make_Stock(737.62, 38.1, 88.9);

```

```

95 Box015 = Make_Stock(874.78, 38.1, 88.9);
96 Cut017 = Make_Cut(Cut019, Sketch019, 0);

```

Listing 4. Figure 9.B - Drafting table (LL-HELM)

```

1 Setup_Tracksaw(0.0000, 0.0000, 9.2500)
2 (@0) = Tracksaw(sheet_0.5x48x96, face_0, edge_0)
3 Setup_Tracksaw(0.0000, 0.0000, 1.0000)
4 (@2,a3) = Tracksaw(@0, face_1, edge_1)
5 Setup_Tracksaw(0.0000, 0.0000, 24.0000)
6 (@4) = Tracksaw(@2, face_2, edge_2)
7 (@5) = Tracksaw(@3, face_3, edge_3)
8 Setup_Chopaw(0.0000, 0.0000, 29.0402)
9 (@6,a7) = Chopaw(lumber_2x4x96, face_4, edge_4)
10 (@9,a10) = Chopaw(lumber_2x4x96, face_5, edge_5)
11 Setup_Chopaw(0.0000, 0.0000, 34.4402)
12 (@11,a12) = Chopaw(@6, face_6, edge_6)
13 (@13) = Chopaw(@9, face_7, edge_7)
14 Setup_Chopaw(20.0000, 0.0000, 11.4462)
15 (@14) = Chopaw(lumber_2x4x24, face_8, edge_8)
16 (@16) = Chopaw(@12, face_9, edge_9)
17 Setup_Chopaw(0.0000, 45.0000, 18.1702)
18 (@17) = Chopaw(lumber_2x4x24, face_10, edge_10)
19 (@18) = Chopaw(lumber_2x4x24, face_11, edge_11)
20 Setup_Chopaw(0.0000, 45.0000, -0.0000)
21 (@19) = Chopaw(@17, face_12, edge_12)
22 (@20) = Chopaw(lumber_2x4x24, face_13, edge_13)
23 (@21) = Chopaw(@20, face_14, edge_14)
24 (@22) = Chopaw(lumber_2x4x24, face_15, edge_15)
25 (@23) = Chopaw(@22, face_16, edge_16)
26 (@24) = Chopaw(@18, face_17, edge_17)
27 Setup_Chopaw(0.0000, 45.0000, 4.4156)
28 (@25) = Chopaw(@21, face_18, edge_18)
29 (@26) = Chopaw(@23, face_19, edge_19)
30 Setup_Chopaw(0.0000, 45.0000, 2.1300)
31 (@27) = Chopaw(@19, face_20, edge_20)
32 (@28) = Chopaw(@24, face_21, edge_21)
33 Return(a4,a5,a7,a10,a11,a13,a14,a16,a25,a26,a27,a28)

```

Listing 5. Figure 9.C - Bookcase (HL-HELM)

```

1 Box001 = Make_Stock(457.2, 38.1, 88.9);
2 MyLine000 = Line(457.2, 0, 435.203, 38.1);
3 Sketch = Make_Sketch(
4     Query_Face_By_Closest_Point(Box001, 228.6, 19.05, 88.9),
5     Geometry(MyLine000),
6     Constraint(Coincident(Start(MyLine000), End(
7         Query_Edge_By_Closest_Point(Box001, 228.6, 0, 88.9))),
8         PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
9             Box001, 228.6, 38.1, 88.9)), Angle(Start(
10            Query_Edge_By_Closest_Point(Box001, 457.2, 19.05, 88.9)), Start(
11            MyLine000), 30)));
12 Cut = Make_Cut(Box001, Sketch, 0);
13 MyLine001 = Line(0, 38.1, 21.997, 0);
14 Sketch001 = Make_Sketch(
15     Query_Face_By_Closest_Point(Cut, 228.6, 19.05, 88.9),
16     Geometry(MyLine001),
17     Constraint(Coincident(Start(MyLine001), End(
18         Query_Edge_By_Closest_Point(Cut, 0, 19.05, 88.9))),
19         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
20             Cut, 228.6, 0, 88.9)), Angle(End(Query_Edge_By_Closest_Point(
21             Cut, 0, 19.05, 88.9)), Start(MyLine001), 30)));
22 Cut001 = Make_Cut(Cut, Sketch001, 1);
23 Box = Make_Stock(457.2, 38.1, 88.9);
24 MyLine002 = Line(435.203, 38.1, 457.2, 0);
25 Sketch004 = Make_Sketch(
26     Query_Face_By_Closest_Point(Box, 228.6, 19.05, 88.9),
27     Geometry(MyLine002),
28     Constraint(Coincident(End(MyLine002), End(
29         Query_Edge_By_Closest_Point(Box, 228.6, 0, 88.9))),
30         PointOnObject(Start(MyLine002), Query_Edge_By_Closest_Point(
31             Box, 228.6, 38.1, 88.9)), Angle(End(Query_Edge_By_Closest_Point(
32             Box, 228.6, 0, 88.9)), -60)));
33 Cut004 = Make_Cut(Box, Sketch004, 0);
34 MyLine003 = Line(0, 38.1, 10.2089, 0);
35 Sketch005 = Make_Sketch(
36     Query_Face_By_Closest_Point(Cut004, 228.6, 19.05, 88.9),
37     Geometry(MyLine003),
38     Constraint(Coincident(Start(MyLine003), End(
39         Query_Edge_By_Closest_Point(Cut004, 0, 19.05, 88.9))),
40         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
41             Cut004, 228.6, 0, 88.9)), Angle(End(Query_Edge_By_Closest_Point(
42             Cut004, 0, 19.05, 88.9)), Start(
43             MyLine003), 15)));
44 Cut005 = Make_Cut(Cut004, Sketch005, 1);
45 Box010 = Make_Stock(152.4, 38.1, 88.9);
46 MyLine004 = Line(152.4, 38.1, 114.3, 0);
47 Sketch013 = Make_Sketch(
48     Query_Face_By_Closest_Point(Box010, 76.2, 19.05, 88.9),
49     Geometry(MyLine004),
50     Constraint(Coincident(Start(MyLine004), End(
51         Query_Edge_By_Closest_Point(Box010, 152.4, 19.05, 88.9))),
52         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
53             Box010, 76.2, 0, 88.9)), Angle(Start(MyLine004), End(
54             Query_Edge_By_Closest_Point(Box010, 152.4, 19.05, 88.9)), 45)));
55 Cut013 = Make_Cut(Box010, Sketch013, 0);
56 Box007 = Make_Stock(304.8, 38.1, 88.9);
57 MyLine005 = Line(304.8, -38.1, 266.7, 0);

```

```

36 Sketch011 = Make_Sketch(
37     Query_Face_By_Closest_Point(Box007, 152.4, 19.05, 0),
38     Geometry(MyLine005),
39     Constraint(Coincident(Start(MyLine005), End(
40         Query_Edge_By_Closest_Point(Box007, 304.8, 19.05, 0)),
41         PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
42             Box007, 152.4, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
43             Box007, 304.8, 19.05, 0)), Start(MyLine005), 45)));
44 Cut011 = Make_Cut(Box007, Sketch011, 0);
45 Box008 = Make_Stock(457.2, 38.1, 88.9);
46 MyLine006 = Line(0, 0, 21.997, 38.1);
47 Sketch002 = Make_Sketch(
48     Query_Face_By_Closest_Point(Box002, 228.6, 19.05, 88.9),
49     Geometry(MyLine006),
50     Constraint(Coincident(Start(MyLine006), Start(Horizontal)),
51         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
52             Box002, 228.6, 38.1, 88.9)), Angle(Start(MyLine006), Start(
53             Query_Edge_By_Closest_Point(Box002, 0, 19.05, 88.9)), 30));
54 Cut002 = Make_Cut(Box002, Sketch002, 1);
55 MyLine007 = Line(457.2, 38.1, 435.203, 0);
56 Sketch003 = Make_Sketch(
57     Query_Face_By_Closest_Point(Cut002, 228.6, 19.05, 88.9),
58     Geometry(MyLine007),
59     Constraint(Coincident(Start(MyLine007), End(
60         Query_Edge_By_Closest_Point(Cut002, 457.2, 19.05, 88.9)),
61         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
62             Cut002, 228.6, 0, 88.9)), Angle(Start(MyLine007), End(
63             Query_Edge_By_Closest_Point(Cut002, 457.2, 19.05, 88.9)),
64             30)));
65 Cut003 = Make_Cut(Cut002, Sketch003, 0);
66 Box008 = Make_Stock(457.2, 38.1, 88.9);
67 Box003 = Make_Stock(609.6, 38.1, 88.9);
68 MyLine008 = Line(609.6, 38.1, 599.391, 0);
69 Sketch006 = Make_Sketch(
70     Query_Face_By_Closest_Point(Box003, 304.8, 19.05, 88.9),
71     Geometry(MyLine008),
72     Constraint(Coincident(Start(MyLine008), End(
73         Query_Edge_By_Closest_Point(Box003, 609.6, 19.05, 88.9)),
74         PointOnObject(End(MyLine008), Query_Edge_By_Closest_Point(
75             Box003, 304.8, 0, 88.9)), Angle(Start(MyLine008), End(
76             Query_Edge_By_Closest_Point(Box003, 609.6, 19.05, 88.9)),
77             15)));
78 Cut006 = Make_Cut(Box003, Sketch006, 0);
79 Box005 = Make_Stock(609.6, -38.1, 587.603, 0);
80 Sketch008 = Make_Sketch(
81     Query_Face_By_Closest_Point(Box005, 304.8, 19.05, 0),
82     Geometry(MyLine009),
83     Constraint(Coincident(Start(MyLine009), End(
84         Query_Edge_By_Closest_Point(Box005, 609.6, 19.05, 0)),
85         PointOnObject(End(MyLine009), Query_Edge_By_Closest_Point(
86             Box005, 304.8, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
87             Box005, 609.6, 19.05, 0)), Start(MyLine009), 30)));
88 Cut008 = Make_Cut(Box005, Sketch008, 0);
89 Box004 = Make_Stock(609.6, 38.1, 88.9);
90 MyLine010 = Line(0, -38.1, 21.997, 0);
91 Sketch007 = Make_Sketch(
92     Query_Face_By_Closest_Point(Box004, 304.8, 19.05, 0),
93     Geometry(MyLine010),
94     Constraint(Coincident(Start(MyLine010), Start(
95         Query_Edge_By_Closest_Point(Box004, 304.8, 38.1, 0)),
96         PointOnObject(End(MyLine010), Query_Edge_By_Closest_Point(
97             Box004, 304.8, 0, 0)), Angle(Start(MyLine010), End(
98             Query_Edge_By_Closest_Point(Box004, 0, 19.05, 0)), 30)));
99 Cut007 = Make_Cut(Box004, Sketch007, 1);
100 Box006 = Make_Stock(609.6, 38.1, 88.9);
101 MyLine011 = Line(0, 38.1, 65.9911, 0);
102 Sketch009 = Make_Sketch(
103     Query_Face_By_Closest_Point(Cut006, 304.8, 19.05, 88.9),
104     Geometry(MyLine011),
105     Constraint(Coincident(Start(MyLine011), End(
106         Query_Edge_By_Closest_Point(Cut006, 0, 19.05, 88.9)),
107         PointOnObject(End(MyLine011), Query_Edge_By_Closest_Point(
108             Cut006, 299.696, 0, 88.9)), Angle(End(
109             Query_Edge_By_Closest_Point(Cut006, 0, 19.05, 88.9)), Start(
110             MyLine011), 60)));
111 Cut009 = Make_Cut(Cut006, Sketch009, 1);
112 Box011 = Make_Stock(152.4, 38.1, 88.9);
113 MyLine012 = Line(152.4, 38.1, 114.3, 0);
114 Sketch014 = Make_Sketch(
115     Query_Face_By_Closest_Point(Box011, 76.2, 19.05, 88.9),
116     Geometry(MyLine012),
117     Constraint(Coincident(Start(MyLine012), End(
118         Query_Edge_By_Closest_Point(Box011, 152.4, 19.05, 88.9)),
119         PointOnObject(End(MyLine012), Query_Edge_By_Closest_Point(
120             Box011, 76.2, 0, 88.9)), Angle(Start(MyLine012), End(
121             Query_Edge_By_Closest_Point(Box011, 152.4, 19.05, 88.9)),
122             45)));
123 Cut014 = Make_Cut(Box011, Sketch014, 0);
124 Box009 = Make_Stock(457.2, 38.1, 88.9);
125 MyLine013 = Line(457.2, 0, 419.1, 38.1);
126 Sketch012 = Make_Sketch(
127     Query_Face_By_Closest_Point(Box009, 228.6, 19.05, 88.9),
128     Geometry(MyLine013),
129     Constraint(Coincident(Start(MyLine013), End(
130         Query_Edge_By_Closest_Point(Box009, 228.6, 0, 88.9)),
131         PointOnObject(End(MyLine013), Query_Edge_By_Closest_Point(
132             Box009, 228.6, 38.1, 88.9)), Angle(Start(
133             Query_Edge_By_Closest_Point(Box009, 457.2, 19.05, 88.9)),
134             Start(MyLine013), 45)));
135 Cut012 = Make_Cut(Box009, Sketch012, 0);

```

```

97 MyLine014 = Line(457.2, 38.1, 435.203, 0);
98 Sketch010 = Make_Sketch(
99     Query_Face_By_Closest_Point(Box008, 228.6, 19.05, 88.9),
100    Geometry(MyLine014),
101    Constraint(Coincident(Start(MyLine014), End(
102        Query_Edge_By_Closest_Point(Box008, 457.2, 19.05, 88.9)),
103        PointOnObject(End(MyLine014), Query_Edge_By_Closest_Point(
104            Box008, 228.6, 0, 88.9)), Angle(Start(MyLine014), End(
105            Query_Edge_By_Closest_Point(Box008, 457.2, 19.05, 88.9)),
106            30)));
107 Cut010 = Make_Cut(Box008, Sketch010, 0);

```

Listing 6. Figure 9.C - Bookcase (LL-HELM)

```

1 Setup_ChopSaw(-45.0000, 0.0000, 12.0000)
2 (a0,a1) = ChopSaw(lumber_2x4x96, face_0, edge_0)
3 Setup_ChopSaw(0.0000, 0.0000, 6.0000)
4 (a3,a4) = ChopSaw(a1, face_1, edge_1)
5 Setup_ChopSaw(-30.0000, 0.0000, 24.0000)
6 (a5,a6) = ChopSaw(lumber_2x4x96, face_2, edge_2)
7 Setup_ChopSaw(-30.0000, 0.0000, 18.0000)
8 (a7,a8) = ChopSaw(a4, face_3, edge_3)
9 (a9,a10) = ChopSaw(a8, face_4, edge_4)
10 (a11,a12) = ChopSaw(a10, face_5, edge_5)
11 Setup_ChopSaw(-45.0000, 0.0000, 18.0000)
12 (a13) = ChopSaw(lumber_2x4x24, face_6, edge_6)
13 Setup_ChopSaw(-15.0000, 0.0000, 18.0000)
14 (a15,a16) = ChopSaw(a6, face_7, edge_7)
15 Setup_ChopSaw(-30.0000, 0.0000, 24.0000)
16 (a17,a18) = ChopSaw(a16, face_8, edge_8)
17 Setup_ChopSaw(0.0000, 0.0000, 24.0000)
18 (a19) = ChopSaw(a12, face_9, edge_9)
19 Setup_ChopSaw(0.0000, 0.0000, 24.2500)
20 (a20,a21) = ChopSaw(a17, face_10, edge_10)
21 Setup_ChopSaw(-45.0000, 0.0000, 6.0000)
22 (a22) = ChopSaw(a21, face_11, edge_11)
23 Setup_ChopSaw(60.0000, 0.0000, 0.0000)
24 (a23) = ChopSaw(a20, face_12, edge_12)
25 Return(a0,a3,a5,a7,a9,a11,a13,a15,a18,a19,a22,a23)

```

Listing 7. Figure 9.D - Birdhouse (HL-HELM)

```

1 Box007 = Make_Stock(426.72, 38.1, 184.15);
2 MyLine000 = Line(-426.72, 184.15, -242.57, 0);
3 Sketch005 = Make_Sketch(
4     Query_Face_By_Closest_Point(Box007, 213.36, 38.1, 92.075),
5     Geometry(MyLine000),
6     Constraint(Coincident(Start(MyLine000), End(
7         Query_Edge_By_Closest_Point(Box007, 213.36, 38.1, 184.15)),
8         PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
9             Box007, 213.36, 38.1, 0)), Angle(End(
10            Query_Edge_By_Closest_Point(Box007, 213.36, 38.1, 0)), End(
11            MyLine000), 135)));
12 Box008 = Make_Stock(426.72, 38.1, 184.15);
13 MyLine001 = Line(-426.72, 184.15, -242.57, 0);
14 Sketch007 = Make_Sketch(
15     Query_Face_By_Closest_Point(Box008, 213.36, 38.1, 92.075),
16     Geometry(MyLine001),
17     Constraint(Coincident(Start(MyLine001), End(
18         Query_Edge_By_Closest_Point(Box008, 213.36, 38.1, 184.15)),
19         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
20             Box008, 213.36, 38.1, 0)), Angle(End(
21            Query_Edge_By_Closest_Point(Box008, 213.36, 38.1, 0)), End(
22            MyLine001), 135)));
23 Box = Make_Stock(152.4, 228.6, 19.05);
24 MyLine002 = Line(0, 152.4, 76.2, 228.6);
25 MyLine003 = Line(76.2, 228.6, 152.4, 152.4);
26 Sketch = Make_Sketch(
27     Query_Face_By_Closest_Point(Box, 76.2, 114.3, 19.05),
28     Geometry(MyLine002, MyLine003),
29     Constraint(PointOnObject(Start(MyLine002), Query_Edge_By_Closest_Point(
30         Box, 0, 114.3, 19.05), PointOnObject(End(MyLine002),
31         Query_Edge_By_Closest_Point(Box, 76.2, 228.6, 19.05)),
32         Coincident(Start(MyLine003), End(MyLine002)), PointOnObject(End(
33             MyLine003), Query_Edge_By_Closest_Point(Box, 152.4, 114.3,
34             19.05)), Angle(End(MyLine003), End(Query_Edge_By_Closest_Point(
35             Box, 152.4, 114.3, 19.05))), 135), Angle(End(
36             Query_Edge_By_Closest_Point(Box, 0, 114.3, 19.05)), Start(
37             MyLine002), 135), Distance(Start(MyLine002), Start(Horizontal),
38             152.4)));
39 Cut = Make_Cut(Box, Sketch, 0);
40 MyCircle004 = Circle(76.2, 139.7, 22.225);
41 Sketch001 = Make_Sketch(
42     Query_Face_By_Closest_Point(Cut, 76.2, 114.3, 19.05),
43     Geometry(MyCircle004),
44     Constraint(DistanceX(MyCircle004, Start(Query_Edge_By_Closest_Point(
45         Cut, 76.2, 114.3, 19.05))), 0, DistanceY(Start(Horizontal),
46         MyCircle004, 139.7)));
47 Cut003 = Make_Cut(Box007, Sketch005, 0);
48 MyLine005 = Line(0, 184.15, -184.15, 0);
49 Sketch006 = Make_Sketch(
50     Query_Face_By_Closest_Point(Cut003, 167.323, 38.1, 92.075),
51     Geometry(MyLine005),

```

```

31     Constraint(Coincident(Start(MyLine005), End(
32         Query_Edge_By_Closest_Point(Cut003, 0, 38.1, 92.075))),
33         PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
34             Cut003, 121.285, 38.1, 0)), Angle(End(
35                 Query_Edge_By_Closest_Point(Cut003, 121.285, 38.1, 0)), End(
36                 MyLine005), 45)));
37 Cut006 = Make_Cut(Box008, Sketch007, 0);
38 MyLine006 = Line(0, 184.15, -184.15, 0);
39 Sketch008 = Make_Sketch(
40     Query_Face_By_Closest_Point(Cut006, 167.323, 38.1, 92.075),
41     Geometry(MyLine006),
42     Constraint(Coincident(Start(MyLine006), End(
43         Query_Edge_By_Closest_Point(Cut006, 0, 38.1, 92.075))),
44         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
45             Cut006, 121.285, 38.1, 0)), Angle(End(
46                 Query_Edge_By_Closest_Point(Cut006, 121.285, 38.1, 0)), End(
47                 MyLine006), 45)));
48 Box003 = Make_Stock(152.4, 228.6, 19.05);
49 MyLine007 = Line(0, 152.4, 76.2, 228.6);
50 MyLine008 = Line(76.2, 228.6, 152.4, 152.4);
51 Sketch002 = Make_Sketch(
52     Query_Face_By_Closest_Point(Box003, 76.2, 114.3, 19.05),
53     Geometry(MyLine007, MyLine008),
54     Constraint(PointOnObject(Start(MyLine007),
55         Query_Edge_By_Closest_Point(Box003, 0, 114.3, 19.05)),
56         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
57             Box003, 76.2, 228.6, 19.05)), Coincident(Start(MyLine008),
58             End(MyLine007)), PointOnObject(End(MyLine008),
59                 Query_Edge_By_Closest_Point(Box003, 152.4, 114.3, 19.05)),
60                 Angle(End(
61                     Query_Edge_By_Closest_Point(Box003, 0, 114.3, 19.05)), Start(
62                     MyLine007), 135), Distance(Start(MyLine007), Start(
63                     Horizontal), 152.4));
64 Cut004 = Make_Cut(Cut003, Sketch006, 1);
65 Hole = Make_Hole(Cut, Sketch001, 44.45);
66 Cut001 = Make_Cut(Box003, Sketch002, 0);
67 Box005 = Make_Stock(133.35, 203.2, 19.05);
68 Box001 = Make_Stock(152.4, 139.7, 19.05);
69 Box002 = Make_Stock(152.4, 139.7, 19.05);
70 Cut005 = Make_Cut(Cut006, Sketch008, 1);
71 Box006 = Make_Stock(152.4, 203.2, 19.05);

```

Listing 8. Figure 9.D - Birdhouse (LL-HELM)

```

19     Query_Face_By_Closest_Point(Cut001, 50.8, 0, 31.75),
20     Geometry(MyArc002, MyArc003, MyArc004, MyLine005),
21     Constraint(Coincident(End(MyArc003), Start(MyArc002)), Coincident(
22         Start(MyLine005), End(MyArc004)), Parallel(MyLine005,
23             Start(MyArc003)), Parallel(MyLine005,
24                 Query_Edge_By_Closest_Point(Cut001, 50.8, 0, 63.5)),
25                 PointOnObject(Start(MyArc004), Query_Edge_By_Closest_Point(
26                     Cut001, 101.6, 0, 31.75)), PointOnObject(End(MyArc002),
27                     Vertical), Distance(MyArc002, Horizontal, 10.16), Distance(
28                     MyArc002, Vertical, 25.4), Distance(MyArc003, Vertical,
29                     41.91), Distance(MyArc003, Horizontal, 28.448), Distance(
30                     MyArc004, Vertical, 77.47));
31 Cut002 = Make_Cut(Cut001, Sketch002, 0);
32 MyCircle006 = Circle(40.64, 35.56, 5.08);
33 Sketch003 = Make_Sketch(
34     Query_Face_By_Closest_Point(Cut002, 52.0243, 0, 23.749),
35     Geometry(MyCircle006),
36     Constraint(Distance(MyCircle006, Query_Edge_By_Closest_Point(Cut002
37         , 0, 0, 9.14598), 40.64), Distance(MyCircle006,
38         Query_Edge_By_Closest_Point(Cut002, 50.8, 0, 0), 35.56));
39 Cut003 = Make_Hole(Cut002, Sketch003, 10.16);
40 MyCircle007 = Circle(40.64, 25.4, 5.08);
41 Sketch004 = Make_Sketch(
42     Query_Face_By_Closest_Point(Cut003, 52.0243, 0, 23.749),
43     Geometry(MyCircle007),
44     Constraint(Distance(MyCircle007, Query_Edge_By_Closest_Point(Cut003
45         , 0, 0, 9.14598), 40.64), Distance(MyCircle007,
46         Query_Edge_By_Closest_Point(Cut003, 50.8, 0, 0), 25.4));
47 Cut004 = Make_Hole(Cut003, Sketch004, 10.16);
48 MyLine008 = Line(45.72, 25.4, 45.72, 35.56);
49 Sketch005 = Make_Sketch(
50     Query_Face_By_Closest_Point(Cut004, 19.1437, 0, 23.749),
51     Geometry(MyLine008),
52     Constraint(PointOnObject(Start(MyLine008),
53         Query_Arc_By_Closest_Center_And_Radius(Cut004, 40.64, 0,
54         25.4, 5.08)), PointOnObject(End(MyLine008),
55         Query_Arc_By_Closest_Center_And_Radius(Cut004, 40.64, 0,
56         35.56, 5.08)));
57 Cut005 = Make_Cut(Cut004, Sketch005, 0);
58 MyLine009 = Line(35.56, 25.4, 35.56, 35.56);
59 Sketch006 = Make_Sketch(
60     Query_Face_By_Closest_Point(Cut005, 19.1437, 0, 23.749),
61     Geometry(MyLine009),
62     Constraint(PointOnObject(End(MyLine009),
63         Query_Arc_By_Closest_Center_And_Radius(Cut005, 40.64, 0,
64         35.56, 5.08)), PointOnObject(Start(MyLine009),
65         Query_Arc_By_Closest_Center_And_Radius(Cut005, 40.64, 0,
66         25.4, 5.08)));
67 Cut006 = Make_Cut(Cut005, Sketch006, 0);

```

Listing 10. Figure 9.E - Toy car (LL-HELM)

```

1 Setup_ChopSaw(0.000, 0.000, 4.000)
2 (a0,a1) = ChopSaw(sheet_0.75x24x20, face_0, edge_0)
3 Setup_ChopSaw(0.000, 0.000, 6.0000)
4 (a3,a4) = ChopSaw(a0, face_1, edge_1)
5 (a5,a6) = ChopSaw(a1, face_2, edge_2)
6 (a7,a8) = ChopSaw(a4, face_3, edge_3)
7 (a9) = ChopSaw(a6, face_4, edge_4)
8 Setup_ChopSaw(0.000, 0.000, 8.000)
9 (a10) = ChopSaw(a7, face_5, edge_5)
10 (a11) = ChopSaw(a8, face_6, edge_6)
11 Setup_ChopSaw(0.000, 0.000, 5.2500)
12 (a12) = ChopSaw(a11, face_7, edge_7)
13 Setup_ChopSaw(-45.0000, 0.0000, 12.0000)
14 (a13,a14) = ChopSaw(a3, face_8, edge_8)
15 Setup_ChopSaw(0.0000, 0.0000, 12.0000)
16 (a15) = ChopSaw(a13, face_9, edge_9)
17 Setup_ChopSaw(45.0000, 0.0000, 6.0000)
18 (a16) = ChopSaw(a14, face_10, edge_10)
19 (a17) = ChopSaw(a15, face_11, edge_11)
20 (a18) = Bandsaw(lumber_2x8x24, face_12, Ref((edge_12, 0, edge_13, 9.76339),
21 (edge_14, 6.98661, edge_15, 0)))
22 (a20) = Bandsaw(a18, face_13, Ref((edge_16, 0, edge_17, 0.0366117), (edge_18,
23 (1.68885, edge_19, 0)))
24 (a21) = Bandsaw(lumber_2x8x24, face_14, Ref((edge_20, 0, edge_21, 9.76339),
25 (edge_22, 6.98661, edge_23, 0)))
26 (a22) = Bandsaw(a21, face_15, Ref((edge_24, 0, edge_25, 0.0366117), (edge_26,
27 (1.68885, edge_27, 0)))
28 Setup_Drill(1.7500)
29 (a23) = Drill(a16, face_16, Ref(edge_28, 3.0000, edge_29, 5.5000))
30 Return(a5,a9,a10,a12,a17,a20,a22,a23)

```

Listing 9. Figure 9.E - Toy car (HL-HELM)

```

1 Box = Make_Stock(101.6, 38.1, 63.5);
2 MyCircle000 = Circle(20.32, 5.08, 1.524);
3 Sketch = Make_Sketch(
4     Query_Face_By_Closest_Point(Box, 50.8, 0, 31.75),
5     Geometry(MyCircle000),
6     Constraint(Distance(MyCircle000, Vertical, 20.32), Distance(
7         MyCircle000, Horizontal, 5.08)));
8 Cut = Make_Hole(Box, Sketch, 3.048);
8 MyCircle001 = Circle(86.36, 5.08, 1.524);
9 Sketch001 = Make_Sketch(
10     Query_Face_By_Closest_Point(Cut, 50.8, 0, 31.75),
11     Geometry(MyCircle001),
12     Constraint(Distance(MyCircle001, Vertical, 86.36), Distance(
13         MyCircle001, Horizontal, 5.08)));
13 Cut001 = Make_Hole(Cut, Sketch001, 3.048);
14 MyArc002 = Arc(25.4, 10.16, 26.67, 1.5933, 2.83175);
15 MyArc003 = Arc(41.91, 28.448, 19.05, 1.5708, 2.68639);
16 MyArc004 = Arc(77.47, 23.368, 24.13, 6.28319, 7.85398);
17 MyLine005 = Line(77.47, 47.498, 41.91, 47.498);
18 Sketch002 = Make_Sketch(

```

```

1 Box009 = Make_Stock(404.774, 38.1, 38.1);
2 Box008 = Make_Stock(404.774, 38.1, 38.1);
3 MyLine000 = Line(0, 38.1, 10.2089, 0);
4 Sketch002 = Make_Sketch(
5     Query_Face_By_Closest_Point(Box009, 202.387, 0, 19.05),
6     Geometry(MyLine000),
7     Constraint(Coincident(Start(MyLine000), End(
8         Query_Edge_By_Closest_Point(Box009, 0, 0, 19.05))), PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
9             Box009, 202.387, 0, 0)), Angle(End(
10            Query_Edge_By_Closest_Point(Box009, 0, 0, 19.05)), Start(
11                MyLine000), 15)));
12 MyLine001 = Line(0, 38.1, 10.2089, 0);
13 Sketch = Make_Sketch(
14     Query_Face_By_Closest_Point(Box008, 202.387, 0, 19.05),
15     Geometry(MyLine001),
16     Constraint(Coincident(Start(MyLine001), End(
17         Query_Edge_By_Closest_Point(Box008, 0, 0, 19.05))), PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
18             Box008, 0, 0, 19.05)), Angle(End(
19                 Query_Edge_By_Closest_Point(Box008, 0, 0, 19.05)), Start(
20                     MyLine001), 15)));
21 Cut003 = Make_Cut(Box009, Sketch002, 1);
22 Cut = Make_Cut(Box008, Sketch, 1);

```

```

15 MyLine002 = Line(404.774, 0, 394.565, 38.1);
16 Sketch003 = Make_Sketch(
17     Query_Face_By_Closest_Point(Cut003, 204.939, 0, 19.05),
18     Geometry(MyLine002),
19     Constraint(Coincident(Start(MyLine002), Start(
20         Query_Edge_By_Closest_Point(Cut003, 404.774, 0, 19.05))),
21         PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
22             Cut003, 202.387, 0, 38.1)), Angle(Start(
23                 Query_Edge_By_Closest_Point(Cut003, 404.774, 0, 19.05)),
24                 Start(MyLine002), 15)));
25 Cut002 = Make_Cut(Cut003, Sketch003, 0);
26 Box016 = Make_Stock(584.2, 38.1, 38.1);
27 Box015 = Make_Stock(152.4, 38.1, 38.1);
28 MyLine003 = Line(0, 38.1, 10.2089, 0);
29 Sketch011 = Make_Sketch(
30     Query_Face_By_Closest_Point(Box015, 76.2, 0, 19.05),
31     Geometry(MyLine003),
32     Constraint(Coincident(Start(MyLine003), End(
33         Query_Edge_By_Closest_Point(Box015, 0, 0, 19.05))),
34         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
35             Box015, 76.2, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
36                 Box015, 0, 0, 19.05)), Start(MyLine003), 15)));
37 Cut013 = Make_Cut(Box015, Sketch011, 1);
38 MyLine004 = Line(152.4, 38.1, 142.191, 0);
39 Sketch012 = Make_Sketch(
40     Query_Face_By_Closest_Point(Cut013, 78.7522, 0, 19.05),
41     Geometry(MyLine004),
42     Constraint(Coincident(Start(MyLine004), End(
43         Query_Edge_By_Closest_Point(Cut013, 76.2, 0, 38.1))),
44         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
45             Cut013, 81.3044, 0, 0)), Angle(Start(MyLine004), End(
46                 Query_Edge_By_Closest_Point(Cut013, 152.4, 0, 19.05)), 15)));
47 Cut014 = Make_Cut(Cut013, Sketch012, 0);
48 Box010 = Make_Stock(101.6, 38.1, 38.1);
49 MyLine005 = Line(0, 38.1, 10.2089, 0);
50 Sketch006 = Make_Sketch(
51     Query_Face_By_Closest_Point(Box010, 50.8, 0, 19.05),
52     Geometry(MyLine005),
53     Constraint(Coincident(Start(MyLine005), End(
54         Query_Edge_By_Closest_Point(Box010, 0, 0, 19.05))),
55         PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
56             Box010, 50.8, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
57                 Box010, 0, 0, 19.05)), Start(MyLine005), 15)));
58 Cut006 = Make_Cut(Box010, Sketch006, 1);
59 MyLine006 = Line(101.6, 0, 79.603, 38.1);
60 Sketch007 = Make_Sketch(
61     Query_Face_By_Closest_Point(Cut006, 53.3522, 0, 19.05),
62     Geometry(MyLine006),
63     Constraint(Coincident(Start(MyLine006), Start(
64         Query_Edge_By_Closest_Point(Cut006, 101.6, 0, 19.05))),
65         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
66             Cut006, 50.8, 0, 38.1)), Angle(Start(
67                 Query_Edge_By_Closest_Point(Cut006, 101.6, 0, 19.05)), Start(
68                 MyLine006), 30)));
69 Cut007 = Make_Cut(Cut006, Sketch007, 0);
70 Box = Make_Stock(381, 38.1, 38.1);
71 Box023 = Make_Stock(101.6, 38.1, 38.1);
72 MyLine007 = Line(0, 38.1, 10.2089, 0);
73 Sketch013 = Make_Sketch(
74     Query_Face_By_Closest_Point(Box023, 50.8, 0, 19.05),
75     Geometry(MyLine007),
76     Constraint(Coincident(Start(MyLine007), End(
77         Query_Edge_By_Closest_Point(Box023, 0, 0, 19.05))),
78         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
79             Box023, 50.8, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
80                 Box023, 0, 0, 19.05)), Start(MyLine007), 15)));
81 Cut015 = Make_Cut(Box023, Sketch013, 1);
82 MyLine008 = Line(-101.6, 0, -79.603, 38.1);
83 Sketch014 = Make_Sketch(
84     Query_Face_By_Closest_Point(Cut015, 53.3522, 38.1, 19.05),
85     Geometry(MyLine008),
86     Constraint(Coincident(Start(MyLine008), Start(
87         Query_Edge_By_Closest_Point(Cut015, 101.6, 38.1, 19.05))),
88         PointOnObject(End(MyLine008), Query_Edge_By_Closest_Point(
89             Cut015, 50.8, 38.1, 38.1)), Angle(Start(MyLine008), Start(
90                 Query_Edge_By_Closest_Point(Cut015, 101.6, 38.1, 19.05)), 30)));
91 Box014 = Make_Stock(152.4, 38.1, 38.1);
92 MyLine009 = Line(152.4, 38.1, 130.403, 0);
93 Sketch010 = Make_Sketch(
94     Query_Face_By_Closest_Point(Box014, 76.2, 0, 19.05),
95     Geometry(MyLine009),
96     Constraint(Coincident(Start(MyLine009), End(
97         Query_Edge_By_Closest_Point(Box014, 76.2, 0, 38.1))),
98         PointOnObject(End(MyLine009), Query_Edge_By_Closest_Point(
99             Box014, 76.2, 0, 0)), Angle(Start(MyLine009), End(
100                 Query_Edge_By_Closest_Point(Box014, 152.4, 0, 19.05)), 30)));
101 Box012 = Make_Stock(101.6, 38.1, 38.1);
102 MyLine010 = Line(0, 38.1, 10.2089, 0);
103 Sketch008 = Make_Sketch(
104     Query_Face_By_Closest_Point(Box012, 50.8, 0, 19.05),
105     Geometry(MyLine010),
106     Constraint(Coincident(Start(MyLine010), End(
107         Query_Edge_By_Closest_Point(Box012, 0, 0, 19.05))),
108         PointOnObject(End(MyLine010), Query_Edge_By_Closest_Point(
109             Box012, 50.8, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
110                 Box012, 0, 0, 19.05)), Start(MyLine010), 15)));
111 Box011 = Make_Stock(404.774, 38.1, 38.1);
112 MyLine011 = Line(0, 38.1, 10.2089, 0);

75 Sketch004 = Make_Sketch(
76     Query_Face_By_Closest_Point(Box011, 202.387, 0, 19.05),
77     Geometry(MyLine011),
78     Constraint(Coincident(Start(MyLine011), End(
79         Query_Edge_By_Closest_Point(Box011, 0, 0, 19.05))),
80         PointOnObject(End(MyLine011), Query_Edge_By_Closest_Point(
81             Box011, 202.387, 0, 0)), Angle(End(
82                 Query_Edge_By_Closest_Point(Box011, 0, 0, 19.05)), Start(
83                 MyLine011), 15)));
84 Box013 = Make_Stock(254, 38.1, 38.1);
85 MyLine012 = Line(0, 38.1, 21.997, 0);
86 Sketch009 = Make_Sketch(
87     Query_Face_By_Closest_Point(Box013, 127, 0, 19.05),
88     Geometry(MyLine012),
89     Constraint(Coincident(Start(MyLine012), End(
90         Query_Edge_By_Closest_Point(Box013, 0, 0, 19.05))),
91         PointOnObject(End(MyLine012), Query_Edge_By_Closest_Point(
92             Box013, 127, 0, 0)), Angle(End(Query_Edge_By_Closest_Point(
93                 Box013, 0, 0, 19.05)), Start(MyLine012), 30)));
94 MyLine013 = Line(404.774, 0, 394.565, 38.1);
95 Sketch001 = Make_Sketch(
96     Query_Face_By_Closest_Point(Cut, 204.939, 0, 19.05),
97     Geometry(MyLine013),
98     Constraint(Coincident(Start(MyLine013), Start(
99         Query_Edge_By_Closest_Point(Cut, 404.774, 0, 19.05))),
100         PointOnObject(End(MyLine013), Query_Edge_By_Closest_Point(
101             Cut, 202.387, 0, 38.1)), Angle(Start(
102                 Query_Edge_By_Closest_Point(Cut, 404.774, 0, 19.05)), Start(
103                 MyLine013), 15)));
104 Cut005 = Make_Cut(Box011, Sketch004, 1);
105 MyLine014 = Line(404.774, 0, 394.565, 38.1);
106 Sketch005 = Make_Sketch(
107     Query_Face_By_Closest_Point(Cut005, 204.939, 0, 19.05),
108     Geometry(MyLine014),
109     Constraint(Coincident(Start(MyLine014), Start(
110         Query_Edge_By_Closest_Point(Cut005, 404.774, 0, 19.05))),
111         PointOnObject(End(MyLine014), Query_Edge_By_Closest_Point(
112             Cut005, 202.387, 0, 38.1)), Angle(Start(
113                 Query_Edge_By_Closest_Point(Cut005, 404.774, 0, 19.05)), Start(
114                 MyLine014), 15)));
115 Box017 = Make_Stock(584.2, 38.1, 38.1);
116 Box019 = Make_Stock(584.2, 38.1, 38.1);
117 Box004 = Make_Stock(457.2, 457.2, 19.05);
118 Box018 = Make_Stock(584.2, 38.1, 38.1);
119 Box022 = Make_Stock(584.2, 38.1, 38.1);
120 Box021 = Make_Stock(584.2, 38.1, 38.1);
121 Cut004 = Make_Cut(Cut005, Sketch005, 0);
122 Cut011 = Make_Cut(Box013, Sketch009, 1);
123 Cut009 = Make_Cut(Box012, Sketch008, 1);
124 Cut001 = Make_Cut(Cut, Sketch001, 0);
125 Cut012 = Make_Cut(Box014, Sketch010, 0);
126 Cut016 = Make_Cut(Cut015, Sketch014, 0);

```

Listing 12. Figure 9.F - Dining room chair (LL-HELM)

```

1 Setup_Tracksaw(0.000, 0.000, 18.000)
2 (a0) = Tracksaw(sheet_0.75x24x20, face_0, edge_0)
3 (a2) = Tracksaw(a0, face_1, edge_1)
4 Setup_Chopaw(-15.000, 0.000, 4.000)
5 (a3,a4) = Chopaw(lumber_2x2x48, face_2, edge_2)
6 Setup_Chopaw(-30.000, 0.000, 6.000)
7 (a6,a7) = Chopaw(lumber_2x2x48, face_3, edge_3)
8 Setup_Chopaw(-30.000, 0.000, 4.000)
9 (a8,a9) = Chopaw(a3, face_4, edge_4)
10 Setup_Chopaw(-15.000, 0.000, 4.000)
11 (a10,a11) = Chopaw(a8, face_5, edge_5)
12 Setup_Chopaw(-15.000, 0.000, 6.000)
13 (a12,a13) = Chopaw(a10, face_6, edge_6)
14 Setup_Chopaw(-15.000, 0.000, 15.875)
15 (a14,a15) = Chopaw(a12, face_7, edge_7)
16 Setup_Chopaw(0.000, 0.000, 10.000)
17 (a16,a17) = Chopaw(a7, face_8, edge_8)
18 Setup_Chopaw(0.000, 0.000, 15.000)
19 (a18) = Chopaw(a14, face_9, edge_9)
20 Setup_Chopaw(0.000, 0.000, 23.000)
21 (a19,a20) = Chopaw(lumber_2x2x48, face_10, edge_10)
22 (a21) = Chopaw(a19, face_11, edge_11)
23 (a22,a23) = Chopaw(lumber_2x2x48, face_12, edge_12)
24 (a24) = Chopaw(a22, face_13, edge_13)
25 (a25) = Chopaw(a17, face_14, edge_14)
26 (a26,a27) = Chopaw(lumber_2x2x48, face_15, edge_15)
27 (a28) = Chopaw(a26, face_16, edge_16)
28 Return(a2,a4,a6,a9,a11,a13,a15,a16,a18,a20,a21,a23,a24,a25,a27,a28)

```

Listing 13. Figure 9.G - Bench (HL-HELM)

```

1 Box = Make_Stock(533.4, 38.1, 88.9);
2 MyLine000 = Line(0, 0, 21.997, 38.1);
3 Sketch002 = Make_Sketch(
4     Query_Face_By_Closest_Point(Box, 266.7, 19.05, 88.9),
5     Geometry(MyLine000),
6     Constraint(Coincident(Start(MyLine000), Start(Horizontal)),
7         PointOnObject(End(MyLine000), Query_Edge_By_Closest_Point(
8             Box, 266.7, 38.1, 88.9)), Angle(Start(
9                 Query_Edge_By_Closest_Point(Box, 266.7, 0, 88.9)), Start(
10                MyLine000), 60)));
11 Cut002 = Make_Cut(Box, Sketch002, 1);
12 MyLine001 = Line(533.4, 0, 511.403, 38.1);

```

```

9 Sketch003 = Make_Sketch(
10     Query_Face_By_Closest_Point(Cut002, 266.7, 19.05, 88.9),
11     Geometry(MyLine001),
12     Constraint(Coincident(Start(MyLine001), End(
13         Query_Edge_By_Closest_Point(Cut002, 266.7, 0, 88.9)),
14         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
15             Cut002, 277.699, 38.1, 88.9)), Angle(Start(MyLine001), End(
16             Query_Edge_By_Closest_Point(Cut002, 266.7, 0, 88.9)), 60)));
17 Cut003 = Make_Cut(Cut002, Sketch003, 0);
18 Box003 = Make_Stock(457.2, 38.1, 88.9);
19 Box001 = Make_Stock(573.02, 38.1, 88.9);
20 MyLine002 = Line(573.02, 0, 551.023, 38.1);
21 Sketch = Make_Sketch(
22     Query_Face_By_Closest_Point(Box001, 286.51, 19.05, 88.9),
23     Geometry(MyLine002),
24     Constraint(Coincident(Start(MyLine002), End(
25         Query_Edge_By_Closest_Point(Box001, 286.51, 0, 88.9)),
26         PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
27             Box001, 286.51, 38.1, 88.9)), Angle(Start(MyLine002), End(
28             Query_Edge_By_Closest_Point(Box001, 286.51, 0, 88.9)), 60)));
29 Cut = Make_Cut(Box001, Sketch, 0);
30 Box006 = Make_Stock(573.02, 38.1, 88.9);
31 MyLine003 = Line(573.02, 0, 551.023, 38.1);
32 Sketch004 = Make_Sketch(
33     Query_Face_By_Closest_Point(Box006, 286.51, 19.05, 88.9),
34     Geometry(MyLine003),
35     Constraint(Coincident(Start(MyLine003), End(
36         Query_Edge_By_Closest_Point(Box006, 286.51, 0, 88.9)),
37         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
38             Box006, 286.51, 38.1, 88.9)), Angle(Start(MyLine003), End(
39             Query_Edge_By_Closest_Point(Box006, 286.51, 0, 88.9)), 60)));
40 Cut004 = Make_Cut(Box006, Sketch004, 0);
41 MyLine004 = Line(529.026, 0, 507.029, 38.1);
42 Sketch009 = Make_Sketch(
43     Query_Face_By_Closest_Point(Cut004, 286.51, 19.05, 88.9),
44     Geometry(MyLine004),
45     Constraint(PointOnObject(Start(MyLine004),
46         Query_Edge_By_Closest_Point(Cut004, 286.51, 0, 88.9)),
47         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
48             Cut004, 275.511, 38.1, 88.9)), Angle(Start(MyLine004), End(
49             Query_Edge_By_Closest_Point(Cut004, 286.51, 0, 88.9)), 60),
50             Distance(Start(MyLine004), End(Query_Edge_By_Closest_Point(
51                 Cut004, 286.51, 0, 88.9)), 43.9941)));
52 Cut008 = Make_Cut(Cut004, Sketch009, 0);
53 Box011 = Make_Stock(573.02, 38.1, 88.9);
54 MyLine005 = Line(573.02, 0, 551.023, 38.1);
55 Sketch013 = Make_Sketch(
56     Query_Face_By_Closest_Point(Box011, 286.51, 19.05, 88.9),
57     Geometry(MyLine005),
58     Constraint(Coincident(Start(MyLine005), End(
59         Query_Edge_By_Closest_Point(Box011, 286.51, 0, 88.9)),
60         PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
61             Box011, 286.51, 38.1, 88.9)), Angle(Start(MyLine005), End(
62             Query_Edge_By_Closest_Point(Box011, 286.51, 0, 88.9)), 60)));
63 Cut012 = Make_Cut(Box011, Sketch013, 0);
64 MyLine006 = Line(485.032, 0, 463.035, 38.1);
65 Sketch021 = Make_Sketch(
66     Query_Face_By_Closest_Point(Cut012, 286.51, 19.05, 88.9),
67     Geometry(MyLine006),
68     Constraint(PointOnObject(Start(MyLine006),
69         Query_Edge_By_Closest_Point(Cut012, 286.51, 0, 88.9)),
70         PointOnObject(End(MyLine006), Query_Edge_By_Closest_Point(
71             Cut012, 275.511, 38.1, 88.9)), Angle(Start(MyLine006), End(
72             Query_Edge_By_Closest_Point(Cut012, 286.51, 0, 88.9)), 60),
73             Distance(End(MyLine006), End(Query_Edge_By_Closest_Point(
74                 Cut012, 562.021, 19.05, 88.9)), 87.9882)));
75 Cut019 = Make_Cut(Cut012, Sketch021, 0);
76 Box027 = Make_Stock(457.2, 38.1, 88.9);
77 Box026 = Make_Stock(457.2, 38.1, 88.9);
78 Box028 = Make_Stock(457.2, 38.1, 88.9);
79 Box021 = Make_Stock(457.2, 38.1, 88.9);
80 Box025 = Make_Stock(457.2, 38.1, 88.9);
81 Box023 = Make_Stock(457.2, 38.1, 88.9);
82 Box024 = Make_Stock(457.2, 38.1, 88.9);
83 Box020 = Make_Stock(573.02, 38.1, 88.9);
84 MyLine007 = Line(573.02, 0, 551.023, 38.1);
85 Sketch022 = Make_Sketch(
86     Query_Face_By_Closest_Point(Box014, 286.51, 19.05, 88.9),
87     Geometry(MyLine007),
88     Constraint(Coincident(Start(MyLine007), End(
89         Query_Edge_By_Closest_Point(Box014, 286.51, 0, 88.9)),
90         PointOnObject(End(MyLine007), Query_Edge_By_Closest_Point(
91             Box014, 286.51, 38.1, 88.9)), Angle(Start(MyLine007), End(
92             Query_Edge_By_Closest_Point(Box014, 286.51, 0, 88.9)), 60)));
93 Cut020 = Make_Cut(Box014, Sketch022, 0);
94 MyLine008 = Line(441.038, 0, 419.041, 38.1);
95 Sketch028 = Make_Sketch(
96     Query_Face_By_Closest_Point(Cut020, 286.51, 19.05, 88.9),
97     Geometry(MyLine008),
98     Constraint(PointOnObject(Start(MyLine008),
99         Query_Edge_By_Closest_Point(Cut020, 286.51, 0, 88.9)),
100         PointOnObject(End(MyLine008), Query_Edge_By_Closest_Point(
101             Cut020, 275.511, 38.1, 88.9)), Angle(Start(MyLine008), End(
102             Query_Edge_By_Closest_Point(Cut020, 286.51, 0, 88.9)), 60),
103             Distance(Start(MyLine008), End(Query_Edge_By_Closest_Point(
104                 Cut020, 286.51, 0, 88.9)), 131.982));
105 Cut026 = Make_Cut(Cut020, Sketch028, 0);
106 Box022 = Make_Stock(457.2, 38.1, 88.9);

```

```

107 Box007 = Make_Stock(573.02, 38.1, 88.9);
108 Box002 = Make_Stock(573.02, 38.1, 88.9);
109 MyLine009 = Line(0, 0, 21.997, 38.1);
110 Sketch005 = Make_Sketch(
111     Query_Face_By_Closest_Point(Box007, 286.51, 19.05, 88.9),
112     Geometry(MyLine009),
113     Constraint(Coincident(Start(MyLine009), Start(Horizontal)),
114         PointOnObject(End(MyLine009), Query_Edge_By_Closest_Point(
115             Box007, 286.51, 38.1, 88.9)), Angle(Start(
116             MyLine009), 60)));
117 Cut005 = Make_Cut(Box007, Sketch005, 1);
118 MyLine010 = Line(43.9941, 0, 65.9911, 38.1);
119 Sketch010 = Make_Sketch(
120     Query_Face_By_Closest_Point(Cut005, 286.51, 19.05, 88.9),
121     Geometry(MyLine010),
122     Constraint(PointOnObject(Start(MyLine010),
123         Query_Edge_By_Closest_Point(Cut005, 286.51, 0, 88.9)), Start(
124             MyLine010), 60, Distance(Start(MyLine010), Start(
125             Horizontal), 43.9941)));
126 Cut009 = Make_Cut(Cut005, Sketch010, 1);
127 Box009 = Make_Stock(209.55, 38.1, 88.9);
128 Box020 = Make_Stock(457.2, 38.1, 88.9);
129 MyLine011 = Line(0, 0, 21.997, 38.1);
130 Sketch011 = Make_Sketch(
131     Query_Face_By_Closest_Point(Box002, 286.51, 19.05, 88.9),
132     Geometry(MyLine011),
133     Constraint(Coincident(Start(MyLine011), Start(Horizontal)),
134         PointOnObject(End(MyLine011), Query_Edge_By_Closest_Point(
135             Box002, 286.51, 38.1, 88.9)), Angle(Start(
136             MyLine011), 60)));
137 Cut001 = Make_Cut(Box002, Sketch011, 1);
138 Box004 = Make_Stock(457.2, 38.1, 88.9);
139 Box015 = Make_Stock(573.02, 38.1, 88.9);
140 MyLine012 = Line(0, 0, 21.997, 38.1);
141 Sketch023 = Make_Sketch(
142     Query_Face_By_Closest_Point(Box015, 286.51, 19.05, 88.9),
143     Geometry(MyLine012),
144     Constraint(Coincident(Start(MyLine012), Start(Horizontal)),
145         PointOnObject(End(MyLine012), Query_Edge_By_Closest_Point(
146             Box015, 286.51, 38.1, 88.9)), Angle(Start(
147             MyLine012), 60)));
148 Cut021 = Make_Cut(Box015, Sketch023, 1);
149 MyLine013 = Line(131.982, 0, 153.979, 38.1);
150 Sketch029 = Make_Sketch(
151     Query_Face_By_Closest_Point(Cut021, 286.51, 19.05, 88.9),
152     Geometry(MyLine013),
153     Constraint(PointOnObject(Start(MyLine013),
154         Query_Edge_By_Closest_Point(Cut021, 286.51, 0, 88.9)),
155         PointOnObject(End(MyLine013), Query_Edge_By_Closest_Point(
156             Cut021, 297.509, 38.1, 88.9)), Angle(Start(
157             MyLine013), 60, Distance(Start(MyLine013), Start(
158             Horizontal), 131.982)));
159 Cut027 = Make_Cut(Cut021, Sketch029, 1);
160 Box012 = Make_Stock(573.02, 38.1, 88.9);
161 MyLine014 = Line(0, 0, 21.997, 38.1);
162 Sketch014 = Make_Sketch(
163     Query_Face_By_Closest_Point(Box012, 286.51, 19.05, 88.9),
164     Geometry(MyLine014),
165     Constraint(Coincident(Start(MyLine014), Start(Horizontal)),
166         PointOnObject(End(MyLine014), Query_Edge_By_Closest_Point(
167             Box012, 286.51, 38.1, 88.9)), Angle(Start(
168             MyLine014), 60)));
169 Cut013 = Make_Cut(Box012, Sketch014, 1);
170 MyLine015 = Line(87.9882, 0, 109.985, 38.1);
171 Sketch020 = Make_Sketch(
172     Query_Face_By_Closest_Point(Cut013, 286.51, 19.05, 88.9),
173     Geometry(MyLine015),
174     Constraint(PointOnObject(Start(MyLine015),
175         Query_Edge_By_Closest_Point(Cut013, 286.51, 0, 88.9)),
176         PointOnObject(End(MyLine015), Query_Edge_By_Closest_Point(
177             Cut013, 297.509, 38.1, 88.9)), Angle(Start(
178             MyLine015), 60, Distance(Start(MyLine015), Start(
179             Horizontal), 87.9882)));
180 Cut018 = Make_Cut(Cut013, Sketch020, 1);
181 Box018 = Make_Stock(573.02, 38.1, 88.9);
182 MyLine016 = Line(0, 0, 21.997, 38.1);
183 Sketch031 = Make_Sketch(
184     Query_Face_By_Closest_Point(Box018, 286.51, 19.05, 88.9),
185     Geometry(MyLine016),
186     Constraint(Coincident(Start(MyLine016), Start(Horizontal)),
187         PointOnObject(End(MyLine016), Query_Edge_By_Closest_Point(
188             Box018, 286.51, 38.1, 88.9)), Angle(Start(
189             MyLine016), 60)));
190 Cut029 = Make_Cut(Box018, Sketch031, 1);
191 MyLine017 = Line(175.976, 0, 197.973, 38.1);
192 Sketch037 = Make_Sketch(
193     Query_Face_By_Closest_Point(Cut029, 286.51, 19.05, 88.9),
194     Geometry(MyLine017),
195     Constraint(PointOnObject(Start(MyLine017),
196         Query_Edge_By_Closest_Point(Cut029, 286.51, 0, 88.9)), Start(
197             MyLine017), 60));

```

```

129     Constraint(PointOnObject(Start(MyLine017),
130         Query_Edge_By_Closest_Point(Cut029, 286.51, 0, 88.9)),
131         PointOnObject(End(MyLine017), Query_Edge_By_Closest_Point(
132             Cut029, 297.509, 38.1, 88.9)), Angle(Start(
133                 Query_Edge_By_Closest_Point(Cut029, 286.51, 0, 88.9)), Start(
134                     MyLine017), 60), Distance(Start(Horizontal), Start(
135                         MyLine017), 175.976));
136 Cut035 = Make_Cut(Cut029, Sketch037, 1);
137 Box017 = Make_Stock(573.02, 38.1, 88.9);
138 MyLine018 = Line(573.02, 0, 551.023, 38.1);
139 Sketch030 = Make_Sketch(
140     Query_Face_By_Closest_Point(Box017, 286.51, 19.05, 88.9),
141     Geometry(MyLine018),
142     Constraint(Coincident(Start(MyLine018), End(
143         Query_Edge_By_Closest_Point(Box017, 286.51, 0, 88.9))),
144         PointOnObject(End(MyLine018), Query_Edge_By_Closest_Point(
145             Box017, 286.51, 38.1, 88.9)), Angle(Start(MyLine018), End(
146             Query_Edge_By_Closest_Point(Box017, 286.51, 0, 88.9)), 60)));
147 Cut028 = Make_Cut(Box017, Sketch030, 0);
148 MyLine019 = Line(397.044, 0, 375.047, 38.1);
149 Sketch036 = Make_Sketch(
150     Query_Face_By_Closest_Point(Cut028, 286.51, 19.05, 88.9),
151     Geometry(MyLine019),
152     Constraint(PointOnObject(Start(MyLine019),
153         Query_Edge_By_Closest_Point(Cut028, 286.51, 0, 88.9)),
154         PointOnObject(End(MyLine019), Query_Edge_By_Closest_Point(
155             Cut028, 275.511, 38.1, 88.9)), Angle(Start(MyLine019), End(
156             Query_Edge_By_Closest_Point(Cut028, 286.51, 0, 88.9)), 60),
157             Distance(Start(MyLine019), End(Query_Edge_By_Closest_Point(
158                 Cut028, 286.51, 0, 88.9)), 175.976)));
159 Cut034 = Make_Cut(Cut028, Sketch036, 0);
160 Box008 = Make_Stock(209.55, 38.1, 88.9);
161 Box019 = Make_Stock(457.2, 38.1, 88.9);
162 Box013 = Make_Stock(533.4, 38.1, 88.9);
163 MyLine020 = Line(0, 0, 21.997, 38.1);
164 Sketch024 = Make_Sketch(
165     Query_Face_By_Closest_Point(Box013, 266.7, 19.05, 88.9),
166     Geometry(MyLine020),
167     Constraint(Coincident(Start(MyLine020), Start(Horizontal)),
168         PointOnObject(End(MyLine020), Query_Edge_By_Closest_Point(
169             Box013, 266.7, 38.1, 88.9)), Angle(Start(
170             Query_Edge_By_Closest_Point(Box013, 266.7, 0, 88.9)), Start(
171                 MyLine020), 60)));
172 Cut022 = Make_Cut(Box013, Sketch024, 1);
173 MyLine021 = Line(533.4, 0, 511.403, 38.1);
174 Sketch025 = Make_Sketch(
175     Query_Face_By_Closest_Point(Cut022, 266.7, 19.05, 88.9),
176     Geometry(MyLine021),
177     Constraint(Coincident(Start(MyLine021), End(
178         Query_Edge_By_Closest_Point(Cut022, 266.7, 0, 88.9))),
179         PointOnObject(End(MyLine021), Query_Edge_By_Closest_Point(
180             Cut022, 277.699, 38.1, 88.9)), Angle(Start(MyLine021), End(
181             Query_Edge_By_Closest_Point(Cut022, 266.7, 0, 88.9)), 60)));
182 Cut023 = Make_Cut(Cut022, Sketch025, 0);
183 MyLine022 = Line(131.982, 0, 153.979, 38.1);
184 Sketch026 = Make_Sketch(
185     Query_Face_By_Closest_Point(Cut023, 266.7, 19.05, 88.9),
186     Geometry(MyLine022),
187     Constraint(PointOnObject(Start(MyLine022),
188         Query_Edge_By_Closest_Point(Cut023, 266.7, 0, 88.9)),
189         PointOnObject(End(MyLine022), Query_Edge_By_Closest_Point(
190             Cut023, 266.7, 38.1, 88.9)), Angle(Start(
191             Query_Edge_By_Closest_Point(Cut023, 266.7, 0, 88.9)), Start(
192                 MyLine022), 60), Distance(Start(MyLine022), Start(Horizontal),
193                 131.982)));
194 Cut024 = Make_Cut(Cut023, Sketch026, 1);
195 MyLine023 = Line(401.418, 0, 379.421, 38.1);
196 Sketch027 = Make_Sketch(
197     Query_Face_By_Closest_Point(Cut024, 332.691, 19.05, 88.9),
198     Geometry(MyLine023),
199     Constraint(PointOnObject(Start(MyLine023),
200         Query_Edge_By_Closest_Point(Cut024, 332.691, 0, 88.9)),
201         PointOnObject(End(MyLine023), Query_Edge_By_Closest_Point(
202             Cut024, 332.691, 38.1, 88.9)), Angle(Start(MyLine023), End(
203             Query_Edge_By_Closest_Point(Cut024, 332.691, 0, 88.9)), 60),
204             Distance(Start(MyLine023), End(Query_Edge_By_Closest_Point(
205                 Cut024, 332.691, 0, 88.9)), 131.982)));
206 Cut025 = Make_Cut(Cut024, Sketch027, 0);
207 Box016 = Make_Stock(533.4, 38.1, 88.9);
208 MyLine024 = Line(0, 0, 21.997, 38.1);
209 Sketch032 = Make_Sketch(
210     Query_Face_By_Closest_Point(Box016, 266.7, 19.05, 88.9),
211     Geometry(MyLine024),
212     Constraint(Coincident(Start(MyLine024), Start(Horizontal)),
213         PointOnObject(End(MyLine024), Query_Edge_By_Closest_Point(
214             Box016, 266.7, 38.1, 88.9)), Angle(Start(
215             Query_Edge_By_Closest_Point(Box016, 266.7, 0, 88.9)), Start(
216                 MyLine024), 60)));
217 Cut030 = Make_Cut(Box016, Sketch032, 1);
218 MyLine025 = Line(533.4, 0, 511.403, 38.1);
219 Sketch033 = Make_Sketch(
220     Query_Face_By_Closest_Point(Cut030, 266.7, 19.05, 88.9),
221     Geometry(MyLine025),
222     Constraint(Coincident(Start(MyLine025), End(
223         Query_Edge_By_Closest_Point(Cut030, 266.7, 0, 88.9))),
224         PointOnObject(End(MyLine025), Query_Edge_By_Closest_Point(
225             Cut030, 266.7, 38.1, 88.9)), Angle(Start(MyLine025), End(
226             Query_Edge_By_Closest_Point(Cut030, 266.7, 0, 88.9)), 60)));
227 Cut034 = Make_Cut(Cut030, Sketch033, 0);
228 MyLine026 = Line(175.976, 0, 197.973, 38.1);
229 Sketch034 = Make_Sketch(
230     Query_Face_By_Closest_Point(Cut034, 266.7, 19.05, 88.9),
231     Geometry(MyLine026),
232     Constraint(Coincident(Start(MyLine026), End(
233         Query_Edge_By_Closest_Point(Cut034, 266.7, 0, 88.9))),
234         PointOnObject(End(MyLine026), Query_Edge_By_Closest_Point(
235             Cut034, 266.7, 38.1, 88.9)), Angle(Start(MyLine026), End(
236             Query_Edge_By_Closest_Point(Cut034, 266.7, 0, 88.9)), 60)));
237 Cut035 = Make_Cut(Cut034, Sketch034, 1);
238 MyLine027 = Line(364.17, 0, 342.173, 38.1);
239 Sketch035 = Make_Sketch(
240     Query_Face_By_Closest_Point(Cut035, 266.7, 19.05, 88.9),
241     Geometry(MyLine027),
242     Constraint(Coincident(Start(MyLine027), End(
243         Query_Edge_By_Closest_Point(Cut035, 266.7, 0, 88.9))),
244         PointOnObject(End(MyLine027), Query_Edge_By_Closest_Point(
245             Cut035, 266.7, 38.1, 88.9)), Angle(Start(MyLine027), End(
246             Query_Edge_By_Closest_Point(Cut035, 266.7, 0, 88.9)), 60)));
247 Cut032 = Make_Cut(Cut031, Sketch034, 1);
248 MyLine028 = Line(533.4, 0, 511.403, 38.1);
249 Sketch036 = Make_Sketch(
250     Query_Face_By_Closest_Point(Cut032, 354.688, 19.05, 88.9),
251     Geometry(MyLine028),
252     Constraint(Coincident(Start(MyLine028), Start(Horizontal)),
253         PointOnObject(End(MyLine028), Query_Edge_By_Closest_Point(
254             Cut032, 354.688, 38.1, 88.9)), Angle(Start(MyLine028), End(
255             Query_Edge_By_Closest_Point(Cut032, 354.688, 0, 88.9)), 60)));
256 Cut033 = Make_Cut(Cut032, Sketch035, 0);
257 Box010 = Make_Stock(533.4, 38.1, 88.9);
258 MyLine029 = Line(0, 0, 21.997, 38.1);
259 Sketch015 = Make_Sketch(
260     Query_Face_By_Closest_Point(Box010, 266.7, 19.05, 88.9),
261     Geometry(MyLine029),
262     Constraint(Coincident(Start(MyLine029), Start(Horizontal)),
263         PointOnObject(End(MyLine029), Query_Edge_By_Closest_Point(
264             Box010, 266.7, 38.1, 88.9)), Angle(Start(
265             Query_Edge_By_Closest_Point(Box010, 266.7, 0, 88.9)), Start(
266                 MyLine029), 60)));
267 Cut014 = Make_Cut(Box010, Sketch015, 1);
268 MyLine029 = Line(533.4, 0, 511.403, 38.1);
269 Sketch016 = Make_Sketch(
270     Query_Face_By_Closest_Point(Cut014, 266.7, 19.05, 88.9),
271     Geometry(MyLine029),
272     Constraint(Coincident(Start(MyLine029), End(
273         Query_Edge_By_Closest_Point(Cut014, 266.7, 0, 88.9))),
274         PointOnObject(End(MyLine029), Query_Edge_By_Closest_Point(
275             Cut014, 277.699, 38.1, 88.9)), Angle(Start(MyLine029), End(
276             Query_Edge_By_Closest_Point(Cut014, 266.7, 0, 88.9)), 60)));
277 Cut015 = Make_Cut(Cut014, Sketch016, 0);
278 MyLine030 = Line(87.9882, 0, 109.985, 38.1);
279 Sketch017 = Make_Sketch(
280     Query_Face_By_Closest_Point(Cut015, 266.7, 19.05, 88.9),
281     Geometry(MyLine030),
282     Constraint(Coincident(Start(MyLine030), End(
283         Query_Edge_By_Closest_Point(Cut015, 266.7, 0, 88.9))),
284         PointOnObject(End(MyLine030), Query_Edge_By_Closest_Point(
285             Cut015, 266.7, 38.1, 88.9)), Angle(Start(
286             Query_Edge_By_Closest_Point(Cut015, 266.7, 0, 88.9)), Start(
287                 MyLine030), 60), Distance(Start(Horizontal), Start(MyLine030),
288                 87.9882)));
289 Cut016 = Make_Cut(Cut015, Sketch017, 1);
290 MyLine031 = Line(423.415, 38.1, 445.412, 0);
291 Sketch018 = Make_Sketch(
292     Query_Face_By_Closest_Point(Cut016, 310.694, 19.05, 88.9),
293     Geometry(MyLine031),
294     Constraint(PointOnObject(Start(MyLine031),
295         Query_Edge_By_Closest_Point(Cut016, 310.694, 38.1, 88.9)),
296         PointOnObject(End(MyLine031), Query_Edge_By_Closest_Point(
297             Cut016, 310.694, 0, 88.9)), Angle(End(MyLine031), End(
298             Query_Edge_By_Closest_Point(Cut016, 310.694, 0, 88.9)), 60),
299             Distance(End(MyLine031), End(Query_Edge_By_Closest_Point(
300                 Cut016, 310.694, 0, 88.9)), 87.9882)));
301 Cut017 = Make_Cut(Cut016, Sketch018, 0);
302 Box005 = Make_Stock(533.4, 38.1, 88.9);
303 MyLine032 = Line(0, 0, 21.997, 38.1);
304 Sketch006 = Make_Sketch(
305     Query_Face_By_Closest_Point(Box005, 266.7, 19.05, 88.9),
306     Geometry(MyLine032),
307     Constraint(Coincident(Start(MyLine032), Start(Horizontal)),
308         PointOnObject(End(MyLine032), Query_Edge_By_Closest_Point(
309             Box005, 266.7, 38.1, 88.9)), Angle(Start(
310             Query_Edge_By_Closest_Point(Box005, 266.7, 0, 88.9)), Start(
311                 MyLine032), 60)));
312 Cut006 = Make_Cut(Box005, Sketch006, 1);
313 MyLine033 = Line(533.4, 0, 511.403, 38.1);
314 Sketch007 = Make_Sketch(
315     Query_Face_By_Closest_Point(Cut006, 266.7, 19.05, 88.9),
316     Geometry(MyLine033),
317     Constraint(Coincident(Start(MyLine033), End(
318         Query_Edge_By_Closest_Point(Cut006, 266.7, 0, 88.9))),
319         PointOnObject(End(MyLine033), Query_Edge_By_Closest_Point(
320             Cut006, 277.699, 38.1, 88.9)), Angle(Start(MyLine033), End(
321             Query_Edge_By_Closest_Point(Cut006, 266.7, 0, 88.9)), 60)));
322 Cut007 = Make_Cut(Cut006, Sketch007, 0);
323 MyLine034 = Line(65.9911, 38.1, 43.9941, 0);
324 Sketch011 = Make_Sketch(
325     Query_Face_By_Closest_Point(Cut007, 266.7, 19.05, 88.9),
326     Geometry(MyLine034),
327     Constraint(PointOnObject(Start(MyLine034),
328         Query_Edge_By_Closest_Point(Cut007, 266.7, 38.1, 88.9)),
329         PointOnObject(End(MyLine034), Query_Edge_By_Closest_Point(
330             Cut007, 266.7, 0, 88.9)), Angle(Start(
331             Query_Edge_By_Closest_Point(Cut007, 266.7, 0, 88.9)), End(
332                 MyLine034), 60), Distance(Start(MyLine034), Start(
333                 Query_Edge_By_Closest_Point(Cut007, 266.7, 38.1, 88.9)),
334                 43.9941)));
335 Cut010 = Make_Cut(Cut007, Sketch011, 1);

```

```

240 MyLine035 = Line(489.406, 0, 467.409, 38.1);
241 Sketch012 = Make_Sketch(
242     Query_Face_By_Closest_Point(Cut010, 288.697, 19.05, 88.9),
243     Geometry(MyLine035),
244     Constraint(PointOnObject(Start(MyLine035),
245         Query_Edge_By_Closest_Point(Cut010, 288.697, 0, 88.9)),
246         PointOnObject(End(MyLine035), Query_Edge_By_Closest_Point(
247             Cut010, 288.697, 38.1, 88.9)), Angle(End(
248             Query_Edge_By_Closest_Point(Cut010, 288.697, 38.1, 88.9)),
249             End(MyLine035), 120), Distance(End(MyLine035), End(
250                 Query_Edge_By_Closest_Point(Cut010, 522.401, 19.05, 88.9)),
251                 43.9941)));
252 Cut011 = Make_Cut(Cut010, Sketch012, 0);

```

Listing 14. Figure 9.G - Bench (LL-HELM)

```

1 Setup_ChopSaw(30.0000, 0.0000, 16.4977)
2 (a0) = ChopSaw(lumber_2x4x24, face_0, edge_0)
3 (a2) = ChopSaw(lumber_2x4x24, face_1, edge_1)
4 (a0) = ChopSaw(lumber_2x4x24, face_2, edge_2)
5 Setup_ChopSaw(30.0000, 0.0000, 34.8633)
6 (a3,a4) = ChopSaw(lumber_2x4x24, face_3, edge_3)
7 Setup_ChopSaw(30.0000, 0.0000, 18.2297)
8 (a6) = ChopSaw(lumber_2x4x24, face_4, edge_4)
9 (a7,a8) = ChopSaw(a4, face_5, edge_5)
10 Setup_ChopSaw(30.0000, 0.0000, 9.7417)
11 (a9,a10) = ChopSaw(a7, face_6, edge_6)
12 Setup_ChopSaw(0.0000, 0.0000, 18.6911)
13 (a11,a12) = ChopSaw(lumber_2x4x24, face_7, edge_7)
14 Setup_ChopSaw(0.0000, 0.0000, 18.0000)
15 (a13) = ChopSaw(lumber_2x4x24, face_8, edge_8)
16 (a14) = ChopSaw(lumber_2x4x24, face_9, edge_9)
17 (a15,a16) = ChopSaw(lumber_2x4x48, face_10, edge_10)
18 (a18) = ChopSaw(a15, face_11, edge_11)
19 (a19) = ChopSaw(lumber_2x4x24, face_12, edge_12)
20 (a20,a21) = ChopSaw(lumber_2x4x48, face_13, edge_13)
21 (a22) = ChopSaw(a20, face_14, edge_14)
22 (a23,a24) = ChopSaw(lumber_2x4x48, face_15, edge_15)
23 (a25) = ChopSaw(lumber_2x4x24, face_16, edge_16)
24 (a26,a27) = ChopSaw(a11, face_17, edge_17)
25 (a28) = ChopSaw(a12, face_18, edge_18)
26 (a29,a30) = ChopSaw(a26, face_19, edge_19)
27 Setup_ChopSaw(0.0000, 0.0000, 31.9457)
28 (a31,a32) = ChopSaw(a3, face_20, edge_20)
29 Setup_ChopSaw(0.0000, 0.0000, 8.2500)
30 (a33,a34) = ChopSaw(a31, face_21, edge_21)
31 (a35) = ChopSaw(a33, face_22, edge_22)
32 Setup_ChopSaw(-30.0000, 0.0000, 21.2500)
33 (a36) = ChopSaw(a33, face_23, edge_23)
34 Setup_ChopSaw(-30.0000, 0.0000, 21.0000)
35 (a37) = ChopSaw(a36, face_24, edge_24)
36 Setup_ChopSaw(-30.0000, 0.0000, 22.5598)
37 (a38) = ChopSaw(lumber_2x4x24, face_25, edge_25)
38 Setup_ChopSaw(-30.0000, 0.0000, 20.8278)
39 (a39,a40) = ChopSaw(a29, face_26, edge_26)
40 Setup_ChopSaw(-30.0000, 0.0000, 15.6316)
41 (a41) = ChopSaw(lumber_2x4x24, face_27, edge_27)
42 (a42,a43) = ChopSaw(a32, face_28, edge_28)
43 Return(a9,a2,a6,a8,a9,a10,a13,a14,a16,a18,a19,a21,a22,a24,a25,a27,a28,a30,a34,a35,
       ,a37,a38,a39,a40,a41,a42,a43)

```

Listing 15. Figure 9.H - Coffee table (HL-HELM)

```

1 Box001 = Make_Stock(914.4, 88.9, 88.9);
2 Box002 = Make_Stock(457.2, 38.1, 88.9);
3 MyLine000 = Line(-38.9584, -1.94e-14, -457.2, 88.9);
4 Sketch = Make_Sketch(
5     Query_Face_By_Closest_Point(Box002, 228.6, 38.1, 44.45),
6     Geometry(MyLine000),
7     Constraint(Coincident(End(MyLine000), End(Query_Edge_By_Closest_Point(
8         Box002, 228.6, 38.1, 88.9))), PointOnObject(Start(MyLine000),
9         Query_Edge_By_Closest_Point(Box002, 228.6, 38.1, 0)), Angle(End(
10        MyLine000), End(Query_Edge_By_Closest_Point(Box002, 228.6,
11            38.1, 88.9)), 12)));
12 Cut001 = Make_Cut(Box002, Sketch, 0);
13 Box012 = Make_Stock(457.2, 38.1, 88.9);
14 Sketch006 = Make_Sketch(
15     Query_Face_By_Closest_Point(Box012, 228.6, 38.1, 44.45),
16     Geometry(MyLine001),
17     Constraint(Coincident(End(MyLine001), End(
18         Query_Edge_By_Closest_Point(Box012, 228.6, 38.1, 88.9))),
19         PointOnObject(Start(MyLine001), Query_Edge_By_Closest_Point(
20             Box012, 228.6, 38.1, 0)), Angle(End(MyLine001), End(
21                 Query_Edge_By_Closest_Point(Box012, 228.6, 38.1, 88.9)), 12));
22 Cut006 = Make_Cut(Box012, Sketch006, 0);
23 Box014 = Make_Stock(457.2, 38.1, 88.9);
24 Box008 = Make_Stock(457.2, 38.1, 88.9);
25 MyLine002 = Line(-457.2, 88.9, -38.9584, 1.42109e-14);
26 Sketch007 = Make_Sketch(
27     Query_Face_By_Closest_Point(Box014, 228.6, 38.1, 44.45),
28     Geometry(MyLine002),

```

```

29     Constraint(Coincident(Start(MyLine002), End(
30         Query_Edge_By_Closest_Point(Box014, 228.6, 38.1, 88.9))), PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
31             Box014, 228.6, 38.1, 0)), Angle(Start(MyLine002), End(
32                 Query_Edge_By_Closest_Point(Box014, 228.6, 38.1, 88.9)), 12));
33 Cut007 = Make_Cut(Box014, Sketch007, 0);
34 MyLine003 = Line(-457.2, 88.9, -38.9584, 1.42109e-14);
35 Sketch003 = Make_Sketch(
36     Query_Face_By_Closest_Point(Box008, 228.6, 38.1, 44.45),
37     Geometry(MyLine003),
38     Constraint(Coincident(Start(MyLine003), End(
39         Query_Edge_By_Closest_Point(Box008, 228.6, 38.1, 88.9))), PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
40             Box008, 228.6, 38.1, 0)), Angle(Start(MyLine003), End(
41                 Query_Edge_By_Closest_Point(Box008, 228.6, 38.1, 88.9)), 12));
42 Cut003 = Make_Cut(Box008, Sketch003, 0);
43 Box = Make_Stock(1828.8, 1828.8, 19.05);
44 MyLine004 = Line(1828.8, 866.506, 1524, 1394.44);
45 MyLine005 = Line(1524, 1394.44, 914.4, 1394.44);
46 MyLine006 = Line(914.4, 1394.44, 689.6, 866.506);
47 MyLine007 = Line(689.6, 866.506, 914.4, 338.577);
48 MyLine008 = Line(914.4, 338.577, 1524, 338.577);
49 MyLine009 = Line(1524, 338.577, 1828.8, 866.506);
50 Sketch001 = Make_Sketch(
51     Query_Face_By_Closest_Point(Box, 914.4, 914.4, 19.05),
52     Geometry(MyLine004, MyLine005, MyLine006, MyLine007, MyLine008,
53         MyLine009),
54     Constraint(Coincident(End(MyLine004), Start(MyLine005)), Coincident(
55         End(MyLine005), Start(MyLine006)), Coincident(End(MyLine006),
56         Start(MyLine007)), Coincident(End(MyLine007), Start(
57             MyLine008)), Coincident(End(MyLine008), Start(MyLine009)),
58             Coincident(End(MyLine009), Start(MyLine004)), Equal(
59                 MyLine004, MyLine005), Equal(MyLine004, MyLine006), Equal(
60                 MyLine004, MyLine007), Equal(MyLine004, MyLine008), Equal(
61                 MyLine004, MyLine009), PointOnObject(End(MyLine009),
62                 Query_Edge_By_Closest_Point(Box, 1828.8, 914.4, 19.05)),
63                 Parallel(MyLine008, Query_Edge_By_Closest_Point(Box, 914.4,
64                     0, 19.05)));
65 Cut = Make_Cut(Box, Sketch001, 1);

```

Listing 16. Figure 9.H - Coffee table (LL-HELM)

```

1 Setup_TrackSaw(0.0000, 0.0000, 6.3058)
2 (a0) = TrackSaw(sheet_0.75x4x96, face_0, edge_0)
3 (a2) = Jigsaw(a0, face_1, Ref((edge_1, 0, edge_2, 36.1972), (edge_3, 35.8028,
4     edge_4, 0)))
4 (a3) = Jigsaw(a2, face_2, Ref((edge_5, 0, edge_6, 20.7221), (edge_5, 20.7221,
5     edge_6, 0)))
6 (a4) = Jigsaw(a3, face_3, Ref((edge_9, 0, edge_10, 20.6931), (edge_11, 0, edge_9,
7     12.0528)))
8 (a5) = Jigsaw(a4, face_4, Ref((edge_13, 12.0528, edge_14, 0), (edge_15, 0,
9     edge_13, 0.0891956)))
10 Setup_ChopSaw(0.0000, 0.0000, 20.1350)
11 (a6,a7) = ChopSaw(lumber_2x4x48, face_5, edge_17)
12 (a9) = ChopSaw(a6, face_6, edge_18)
13 Setup_ChopSaw(0.0000, 0.0000, 36.0000)
14 (a10) = ChopSaw(lumber_4x4x48, face_7, edge_19)
15 (a12,a13) = Bandsaw(a9, face_8, Ref((edge_20, 0, edge_21, 1.8344), (edge_22, 0,
16     edge_23, 1.8344)))
17 (a14,a15) = Bandsaw(a7, face_9, Ref((edge_24, 0, edge_25, 1.8344), (edge_26,
18     1.8344, edge_27, 0)))
19 Return(a5,a10,a12,a13,a14,a15)

```

Listing 17. Figure 9.I - Flower pot (HL-HELM)

```

1 Box009 = Make_Stock(152.4, 152.4, 12.7);
2 MyLine000 = Line(87.9882, 0, 0, -152.4);
3 Sketch006 = Make_Sketch(
4     Query_Face_By_Closest_Point(Box009, 76.2, 76.2, 0),
5     Geometry(MyLine000),
6     Constraint(PointOnObject(Start(MyLine000),
7         Query_Edge_By_Closest_Point(Box009, 76.2, 0, 0)), Angle(End(
8             MyLine000), End(Query_Edge_By_Closest_Point(Box009, 0, 76.2,
9                 0)), 30), Coincident(End(MyLine000), Start(
10                Query_Edge_By_Closest_Point(Box009, 76.2, 152.4, 0))));
11 Cut006 = Make_Cut(Box009, Sketch006, 1);
12 Box002 = Make_Stock(127, 38.1, 38.1);
13 Box014 = Make_Stock(127, 38.1, 38.1);
14 Box012 = Make_Stock(304.8, 25.4, 12.7);
15 MyLine001 = Line(-25.4, 12.7, -18.0677, 0);
16 Sketch007 = Make_Sketch(
17     Query_Face_By_Closest_Point(Box012, 0, 12.7, 6.35),
18     Geometry(MyLine001),
19     Constraint(Coincident(Start(MyLine001), End(
20         Query_Edge_By_Closest_Point(Box012, 0, 25.4, 6.35))),
21         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
22             Box012, 0, 12.7, 0)), Angle(End(Query_Edge_By_Closest_Point(
23                 Box012, 0, 25.4, 6.35)), Start(MyLine001), 30));
24 Cut007 = Make_Cut(Box012, Sketch007, 0);
25 Box010 = Make_Stock(330.2, 152.4, 12.7);
26 Box013 = Make_Stock(152.4, 152.4, 12.7);
27 MyLine002 = Line(87.9882, 0, 0, -152.4);
28 Sketch008 = Make_Sketch(
29     Query_Face_By_Closest_Point(Box013, 76.2, 76.2, 0),
30     Geometry(MyLine002),

```

```

23     Constraint(PointOnObject(Start(MyLine002),
24         Query_Edge_By_Closest_Point(Box013, 76.2, 0, 0)), Angle(End(
25             MyLine002), End(Query_Edge_By_Closest_Point(Box013, 0, 76.2,
26                 0)), 30), Coincident(End(MyLine002), Start(
27                 Query_Edge_By_Closest_Point(Box013, 76.2, 152.4, 0))));;
28 Cut008 = Make_Cut(Box013, Sketch008, 1);
29 Box004 = Make_Stock(330.2, 203.2, 12.7);
30 MyLine003 = Line(330.2, 177.8, 0, 177.8);
31 Sketch001 = Make_Sketch(
32     Query_Face_By_Closest_Point(Box004, 165.1, 101.6, 12.7),
33     Geometry(MyLine003),
34     Constraint(PointOnObject(Start(MyLine003),
35         Query_Edge_By_Closest_Point(Box004, 330.2, 101.6, 12.7)),
36         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
37             Box004, 0, 101.6, 12.7)), Distance(Start(Horizontal),
38             MyLine003, 177.8), Distance(Start(MyLine003), End(
39                 Query_Edge_By_Closest_Point(Box004, 165.1, 0, 12.7)), 177.8));
40 Cut001 = Make_Cut(Box004, Sketch001, 0);
41 MyLine004 = Line(-177.8, -5.08e-13, -170.468, 12.7);
42 Sketch004 = Make_Sketch(
43     Query_Face_By_Closest_Point(Cut001, 0, 88.9, 6.35),
44     Geometry(MyLine004),
45     Constraint(Coincident(Start(MyLine004), End(
46         Query_Edge_By_Closest_Point(Cut001, 0, 88.9, -2.615e-13)),
47         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
48             Cut001, 0, 88.9, 12.7)), Angle(Start(MyLine004), End(
49             Query_Edge_By_Closest_Point(Cut001, 0, 177.8, 6.35)), 30)));
50 Cut004 = Make_Cut(Cut001, Sketch004, 0);
51 Box011 = Make_Stock(91.44, 330.2, 12.7);

```

Listing 18. Figure 9.I - Flower pot (LL-HELM)

```

1 Setup_Tracksaw(0.0000, 0.0000, 20.2750)
2 (a0,a1) = Tracksaw(sheet_.0x5x24x20, face_0, edge_0)
3 Setup_Tracksaw(0.0000, 0.0000, 6.0000)
4 (a3,a4) = Tracksaw(a0, face_1, edge_1)
5 Setup_Tracksaw(0.0000, 0.0000, 6.1693)
6 (a5) = Tracksaw(a3, face_2, edge_2)
7 Setup_Tracksaw(0.0000, 0.0000, 13.0000)
8 (a6) = Tracksaw(a1, face_3, edge_3)
9 (a7,a8) = Tracksaw(a4, face_4, edge_4)
10 (a9) = Tracksaw(a5, face_5, edge_5)
11 Setup_Tracksaw(0.0000, -30.0000, 6.7113)
12 (a10,a11) = Tracksaw(a9, face_6, edge_6)
13 Setup_Chopsw(30.0000, 0.0000, 6.0000)
14 (a12,a13) = Chopsw(a8, face_7, edge_7)
15 Setup_Chopsw(0.0000, 0.0000, 12.0000)
16 (a14) = Chopsw(a10, face_8, edge_8)
17 Setup_Chopsw(0.0000, 0.0000, 6.0000)
18 (a15) = Chopsw(a12, face_9, edge_9)
19 Setup_Chopsw(0.0000, 0.0000, 5.0000)
20 (a16,a17) = Chopsw(lumber_2x2x24, face_10, edge_10)
21 (a19) = Chopsw(a16, face_11, edge_11)
22 Return(a6,a7,a11,a13,a14,a15,a17,a19)

```

Listing 19. Figure 9.J - Z-table (HL-HELM)

```

1 Box001 = Make_Stock(792.48, 38.1, 88.9);
2 MyLine000 = Line(689.813, 88.9, 638.487, 0);
3 Sketch = Make_Sketch(
4     Query_Face_By_Closest_Point(Box001, 396.24, 0, 44.45),
5     Geometry(MyLine000),
6     Constraint(PointOnObject(Start(MyLine000), Query_Edge_By_Closest_Point(
7         Box001, 396.24, 0, 88.9)), PointOnObject(End(MyLine000),
8         Query_Edge_By_Closest_Point(Box001, 396.24, 0, 0)), Angle(End(
9             MyLine000), End(Query_Edge_By_Closest_Point(Box001, 396.24, 0,
10                0)), 120), Distance(Start(MyLine000), End(
11                 Query_Edge_By_Closest_Point(Box001, 396.24, 0, 88.9)), 102.667));
12 Cut = Make_Cut(Box001, Sketch, 0);
13 MyLine001 = Line(0, 0, 51.3264, 88.9);
14 Sketch002 = Make_Sketch(
15     Query_Face_By_Closest_Point(Cut, 332.075, 0, 44.45),
16     Geometry(MyLine001),
17     Constraint(Coincident(Start(MyLine001), Start(Horizontal)),
18         PointOnObject(End(MyLine001), Query_Edge_By_Closest_Point(
19             Cut, 344.907, 0, 88.9)), Angle(Start(
20                 Query_Edge_By_Closest_Point(Cut, 319.243, 0, 0)), Start(
21                 MyLine001), 60)));
22 Cut002 = Make_Cut(Cut, Sketch002, 1);
23 Box = Make_Stock(533.4, 38.1, 88.9);
24 MyLine002 = Line(533.4, 0, 482.074, 88.9);
25 Sketch001 = Make_Sketch(
26     Query_Face_By_Closest_Point(Box, 266.7, 0, 44.45),
27     Geometry(MyLine002),
28     Constraint(Coincident(Start(MyLine002), Start(
29         Query_Edge_By_Closest_Point(Box, 533.4, 0, 44.45))),
30         PointOnObject(End(MyLine002), Query_Edge_By_Closest_Point(
31             Box, 266.7, 0, 88.9)), Angle(End(Query_Edge_By_Closest_Point(
32                 Box, 266.7, 0, 88.9)), End(MyLine002), 120));
33 Cut001 = Make_Cut(Box, Sketch001, 0);
34 Box013 = Make_Stock(914.4, 38.1, 88.9);
35 Box014 = Make_Stock(914.4, 86.36, 12.7);
36 Box003 = Make_Stock(792.48, 38.1, 88.9);
37 MyLine003 = Line(689.813, 88.9, 638.487, 0);
38 Sketch003 = Make_Sketch(
39     Query_Face_By_Closest_Point(Box003, 396.24, 0, 44.45),

```

```

27     Geometry(MyLine003),
28     Constraint(PointOnObject(Start(MyLine003),
29         Query_Edge_By_Closest_Point(Box003, 396.24, 0, 88.9)),
30         PointOnObject(End(MyLine003), Query_Edge_By_Closest_Point(
31             Box003, 396.24, 0, 0)), Angle(End(MyLine003), End(
32                 Query_Edge_By_Closest_Point(Box003, 396.24, 0, 0)), 120),
33                 Distance(Start(MyLine003), End(Query_Edge_By_Closest_Point(
34                     Box003, 396.24, 0, 88.9)), 102.667));
35 Cut003 = Make_Cut(Box003, Sketch003, 0);
36 MyLine004 = Line(0, 0, 51.3264, 88.9);
37 Sketch004 = Make_Sketch(
38     Query_Face_By_Closest_Point(Cut003, 332.075, 0, 44.45),
39     Geometry(MyLine004),
40     Constraint(Coincident(Start(MyLine004), Start(Horizontal)),
41         PointOnObject(End(MyLine004), Query_Edge_By_Closest_Point(
42             Cut003, 344.907, 0, 88.9)), Angle(Start(
43                 Query_Edge_By_Closest_Point(Cut003, 319.243, 0, 0)), Start(
44                 MyLine004), 60)));
45 Cut005 = Make_Cut(Cut003, Sketch004, 1);
46 Box011 = Make_Stock(495.3, 914.4, 12.7);
47 Box012 = Make_Stock(914.4, 38.1, 88.9);
48 Box008 = Make_Stock(495.3, 914.4, 12.7);
49 Box016 = Make_Stock(533.4, 38.1, 88.9);
50 Box009 = Make_Stock(533.4, 63.5, 12.7);
51 Box015 = Make_Stock(533.4, 38.1, 88.9);
52 Box017 = Make_Stock(365.76, 63.5, 12.7);
53 Box018 = Make_Stock(365.76, 63.5, 12.7);
54 Box002 = Make_Stock(533.4, 38.1, 88.9);
55 MyLine005 = Line(533.4, 0, 482.074, 88.9);
56 Sketch005 = Make_Sketch(
57     Query_Face_By_Closest_Point(Box002, 266.7, 0, 44.45),
58     Geometry(MyLine005),
59     Constraint(Coincident(Start(MyLine005), Start(
60                 Query_Edge_By_Closest_Point(Box002, 533.4, 0, 44.45))),
61                 PointOnObject(End(MyLine005), Query_Edge_By_Closest_Point(
62                     Box002, 266.7, 0, 88.9)), Angle(End(
63                     Query_Edge_By_Closest_Point(Box002, 266.7, 0, 88.9)), End(
64                     MyLine005), 120));
65 Cut004 = Make_Cut(Box002, Sketch005, 0);
66 Box010 = Make_Stock(495.3, 63.5, 12.7);

```

Listing 20. Figure 9.J - Z-table (LL-HELM)

```

1 Setup_Tracksaw(0.0000, 0.0000, 36.0000)
2 (a0,a1) = Tracksaw(sheet_.0x5x48x96, face_0, edge_0)
3 Setup_Tracksaw(0.0000, 0.0000, 23.0250)
4 (a3,a4) = Tracksaw(a0, face_1, edge_1)
5 Setup_Tracksaw(0.0000, 0.0000, 2.5000)
6 (a5,a6) = Tracksaw(a1, face_2, edge_2)
7 (a7) = Tracksaw(a6, face_3, edge_3)
8 Setup_Tracksaw(0.0000, 0.0000, 19.5000)
9 (a8) = Tracksaw(a3, face_4, edge_4)
10 (a9,a10) = Tracksaw(a4, face_5, edge_5)
11 (a11,a12) = Tracksaw(a5, face_6, edge_6)
12 (a13,a14) = Tracksaw(a7, face_7, edge_7)
13 Setup_Chopsw(30.0000, 0.0000, -0.0000)
14 (a15) = Chopsw(lumber_2x4x96, face_8, edge_8)
15 Setup_Chopsw(30.0000, 0.0000, 25.1373)
16 (a17,a18) = Chopsw(a18, face_9, edge_9)
17 (a19) = Chopsw(a18, face_10, edge_10)
18 Setup_Chopsw(0.0000, 0.0000, 14.4000)
19 (a20) = Chopsw(a11, face_11, edge_11)
20 (a21) = Chopsw(a13, face_12, edge_12)
21 Setup_Chopsw(0.0000, 0.0000, 36.0000)
22 (a22) = Chopsw(lumber_2x4x48, face_13, edge_13)
23 (a24,a25) = Chopsw(lumber_2x4x96, face_14, edge_14)
24 Setup_Chopsw(-30.0000, 0.0000, 21.0000)
25 (a26,a27) = Chopsw(lumber_2x4x48, face_15, edge_15)
26 Setup_Chopsw(0.0000, 0.0000, 21.0000)
27 (a28) = Chopsw(a27, face_16, edge_16)
28 (a29,a30) = Chopsw(a24, face_17, edge_17)
29 (a31) = Chopsw(a29, face_18, edge_18)
30 Return(a8,a9,a10,a12,a14,a17,a19,a20,a21,a22,a25,a26,a28,a30,a31)

```