Test Plan Document

Lumber One

Steven Moore

CS 4321 - Software Engineering I

21 November, 2016

Use-Case

- Input Log Data:
  - Inputs:
    - (A)-Entries per log: {<3,3,>3}
    - (B)-Quantity of logs: {<=0,>0}
    - (C)-Log dimensions: {<=0,>0}

- Input Lumber Data:
  - Inputs:
    - (A)-Entries per lumber: {<4,4,>4}
    - (B)-Quantity of lumber: {<=0,>0}
    - (C)-Lumber dimensions: {<=0,>0}
    - (D)-Lumber value: {<=0,>0}
    - (E)-Quantity of scrap: {<1,1,>1}
    - (F)-Scrap value: {<=0,>0}

- Cut Logs:
  - Inputs:
    - (A)-Log data: {exists,DNE}
    - (B)-Lumber data: {exists,DNE}
    - (C)-Quantity of Logs: {>0}
    - (D)-Log dimensions (L/W/H): {>0,>0,>0}
    - (E)-Quantity of lumber: {>0}
    - (F)-Lumber dimensions (L/W/H): {>0,>0,>0}
    - (G)-Lumber Value: {>0}
    - (H)-Scrap value: {>0}

- Lumber Sold:
  - Inputs:
    - (A)-Address cheap rate: {exists,DNE}
    - (B)-Purchase lumber: {exists,DNE}

Input Log Data Test

Test	Specification	Test Case	Expected Results
1	A:[<3] B:[>0] C:[>0]	A=2 B=1 C=1	Invalid Input (A)
2	A:[>3] B:[>0] C:[>0]	A=4 B=1 C=1	Invalid Input (A)
3	A:[3] B:[<=0] C:[>0]	A=3 B=0 C=1	Invalid Input (B)
4	A:[3] B:[>0] C:[<=0]	A=3 B=1 C=0	Invalid Input (C)
5	A:[3] B:[>0] C:[>0]	A=3 B=1 C=1	(1) Log with L/W/H of 1/1/1

Test	Purpose
1	Show that log file with less than 3 entries per log will not be accepted.
2	Show that log file with greater than 3 entries per log will not be accepted.
3	Show that log file with no logs will not be accepted.
4	Show that log file with log dimensional values of 0 or less will not be accepted.
5	Show that log file with acceptable log parameters will be accepted.

Generic "Input Log Data Test" Directions:

1. Run the system
2. Choose, “Upload Log Data” from the File menu.
3. Navigate to and enter the Log Data with information pertaining to specific "Input Log Data Test" by uploading *log_testcase_xx.txt.*
4. Verify output matches expected results.
5. Perform steps 2-4 for each test in the "Input Log Data Test" set.
6. Exit the system.

Input Lumber Data Test

Test	Specification	Test Case	Expected Results
1	A:[<4] B:[>0] C:[>0] D:[>0] E:[1] F:[>0]	A=3 B=1 C=2 D=8 E=1 F=1	Invalid Input (A)
2	A:[>4] B:[>0] C:[>0] D:[>0] E:[1] F:[>0]	A=5 B=1 C=2 D=8 E=1 F=1	Invalid Input (A)
3	A:[4] B:[<=0] C:[>0] D:[>0] E:[1] F:[>0]	A=4 B=0 C=2 D=8 E=1 F=1	Invalid Input (B)
4	A:[4] B:[>0] C:[<=0] D:[>0] E:[1] F:[>0]	A=4 B=1 C=0 D=8 E=1 F=1	Invalid Input (C)
5	A:[4] B:[>0] C:[>0] D:[<=0] E:[1] F:[>0]	A=4 B=1 C=2 D=0 E=1 F=1	Invalid Input (D)
6	A:[4] B:[>0] C:[>0] D:[>0] E:[<1] F:[>0]	A=4 B=1 C=2 D=8 E=0 F=1	Invalid Input (E)
7	A:[4] B:[>0] C:[>0] D:[>0] E:[>1] F:[>0]	A=4 B=1 C=2 D=8 E=2 F=1	Invalid Input (E)
8	A:[4] B:[>0] C:[>0] D:[>0] E:[1] F:[<=0]	A=4 B=1 C=2 D=8 E=1 F=0	Invalid Input (F)
9	A:[4] B:[>0] C:[>0] D:[>0] E:[1] F:[>0]	A=4 B=1 C=2 D=8 E=1 F=1	(1) Lumber: L/W/H: 2/2/2 Value: 8 (1) Scrap: Value: 1

Test	Purpose
1	Show that lumber file with less than 4 entries per lumber will not be accepted.
2	Show that lumber file with greater than 4 entries per lumber will not be accepted.
3	Show that lumber file with no lumber entries will not be accepted.
4	Show that lumber file with lumber dimensional values of 0 or less will not be accepted.
5	Show that lumber file with lumber value of 0 or less will not be accepted.
6	Show that lumber file with no scrap will not be accepted.
7	Show that lumber file with multiple scrap values will not be accepted.
8	Show that lumber file with scrap value of 0 or less will not be accepted.
9	Show that lumber file with acceptable parameters will be accepted.

Generic "Input Lumber Data Test" Directions:

1. Run the system.
2. Choose, “Upload Lumber Data” from the File menu.
3. Navigate to and enter the Lumber Data with information pertaining to specific "Input Lumber Data Test" by uploading *lumber_testcase_xx.txt.*
4. Verify output matches expected results.
5. Perform steps 2-4 for each test in the "Input Lumber Data Test" set.
6. Exit the system.

Cut Logs Test

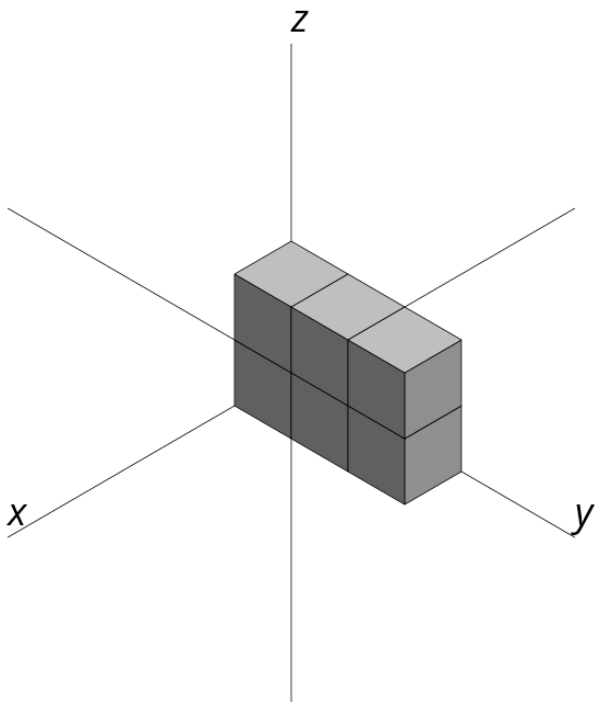
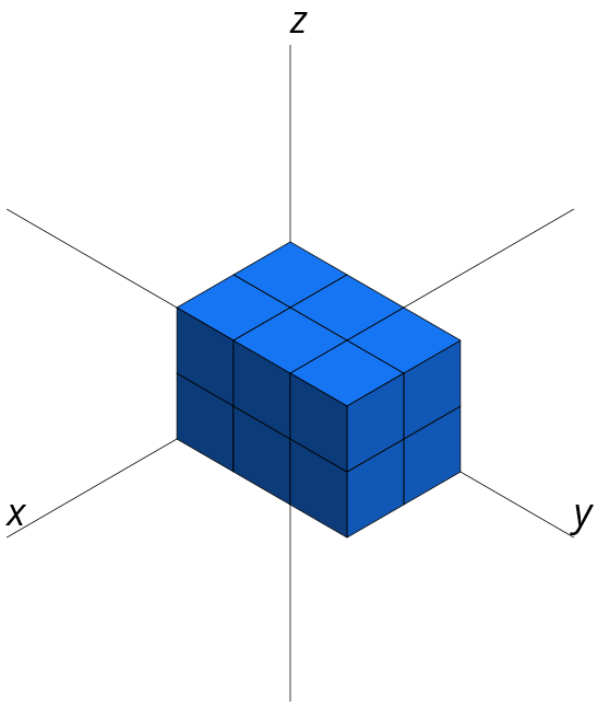
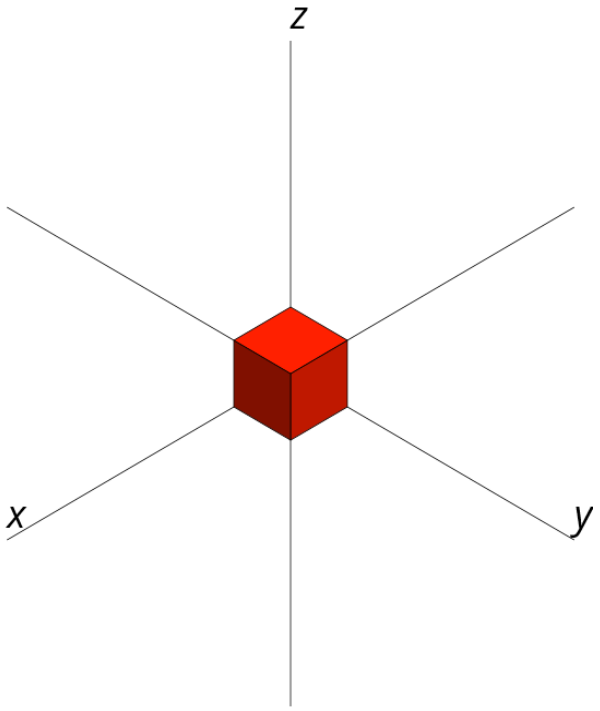
Test	Specification	Test Case	Expected Results
1	A:[DNE] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=false B=true C=1 D=1/1/1 E=1 F=1/1/1 G=1 H=0.1	Invalid Input (A)
2	A:[exists] B:[DNE] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=false C=1 D=1/1/1 E=1 F=1/1/1 G=1 H=0.1	Invalid Input (B)
3	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/1 E=1 F=2/3/2 G=1 H=0.1	See Explanation
4	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/1 E=1 F=2/3/1 G=1 H=0.1	See Explanation
5	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/1 E=1 F=2/2/1 G=1 H=0.1	See Explanation
6	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/5 E=1 F=1/3/5 G=1 H=0.1	See Explanation
7	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=3/2/5 E=1 F=1/3/5 G=1 H=0.1	See Explanation
8	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/5 E=1 F=3/1/5 G=1 H=0.1	See Explanation
9	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/5 E=1 F=1/3/5 G=1 H=1	See Explanation
10	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/5 E=2 F=1/3/5 G=3 H=0.1 F=1/2/5 G=2	See Explanation
11	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/3/5 E=2 F=1/3/5 G=2 H=0.1 F=1/2/5 G=3	See Explanation
12	A:[exists] B:[exists] C:[>0] D:[>0,>0,>0] E:[>0] F:[>0,>0,>0] G:[>0] H:[>0]	A=true B=true C=1 D=2/4/5 E=2 F=2/3/5 G=3 H=0.1 F=2/2/5 G=2	See Explanation

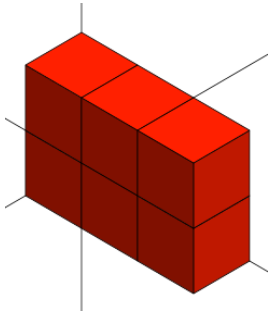
Test	Purpose
1	Show that cutting with no logs will not be accepted.
2	Show that cutting with no lumber will not be accepted.
3	Show what happens when attempting to cut when length of log is less than length of lumber.
4	Show what happens when attempting to cut when length of log is equal to length of lumber.
5	Show what happens when attempting to cut when length of log is greater than length of lumber.
6	Show what happens when attempting to cut when length of log is double length of lumber, also used for comparison against next two test cases.
7	Show that output is the same as test 6 when switching length/width (rotating) log.
8	Show that output is the same as test 6 when switching length/width (rotating) lumber.
9	Show what happens when cutting with scrap having a greater value than lumber.
10	Show standard cut for one log versus multiple lumber, used for comparison against next two test cases.
11	Show what happens when cutting with smaller lumber cut has greater value than larger cut.
12	Show that algorithm is brute force and not greedy method.

Generic "Cut Logs Test" Directions:

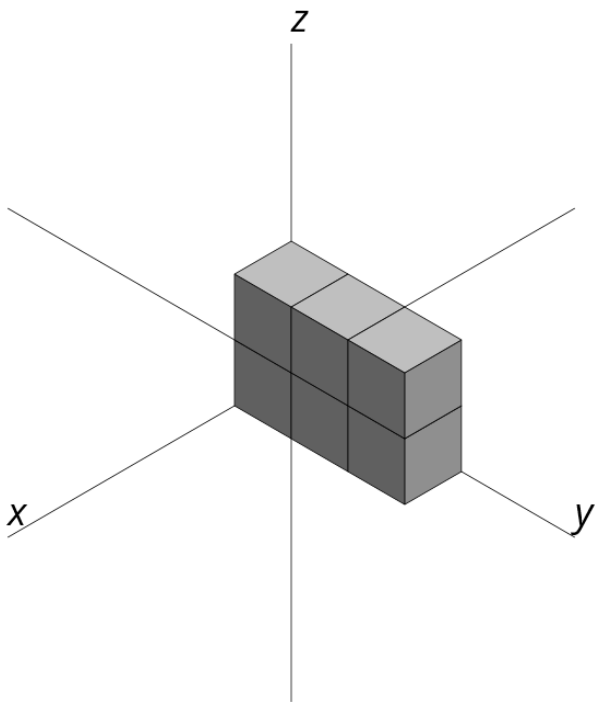
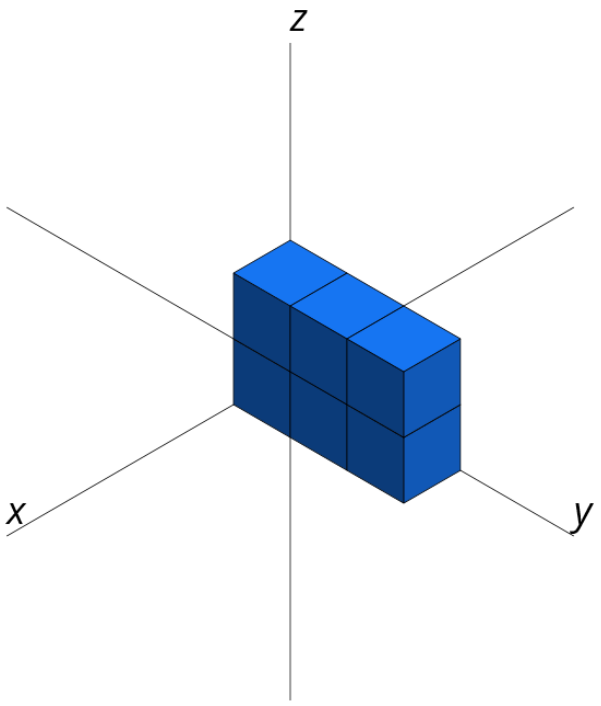
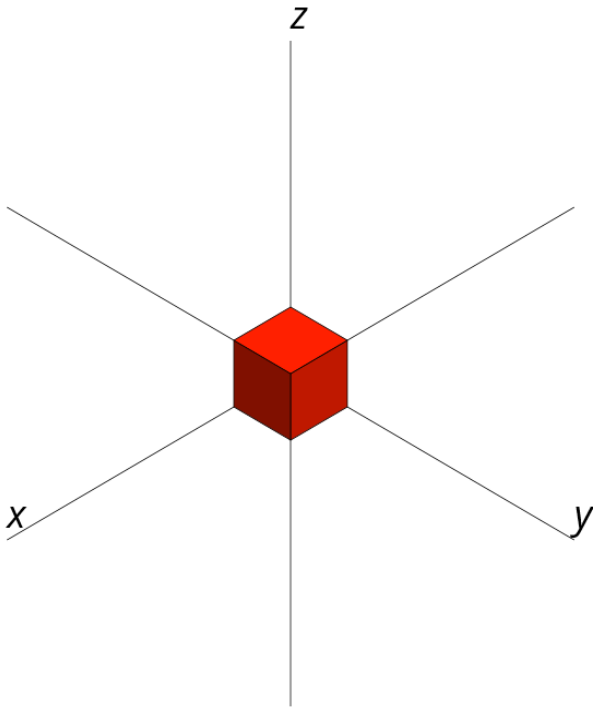
1. Run the system.
2. Upload log data & lumber data.
3. Choose: “Cut Logs” from the file menu.
4. Verify output matches expected results.
5. Perform steps 2-5 for each test in the "Cut Logs Test" set.
6. Exit the system.

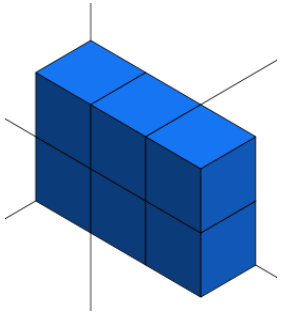
Test 3

Log: 2x3x1	Lumber: 2x3x2 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

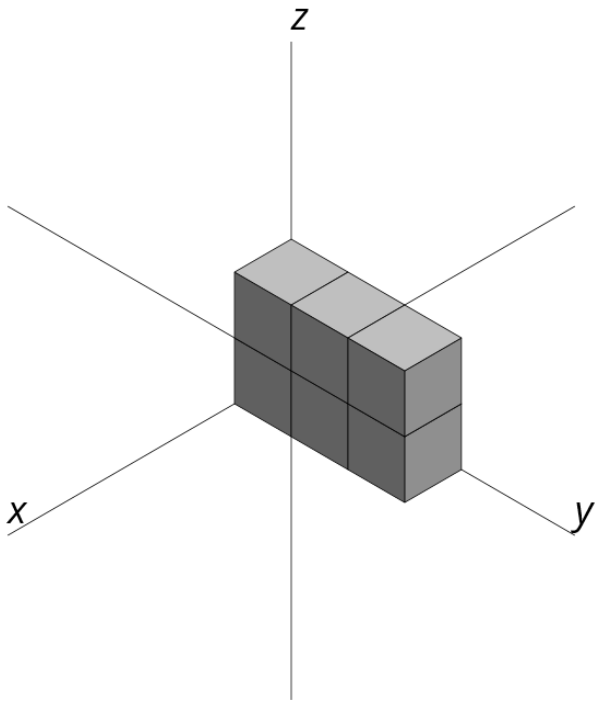
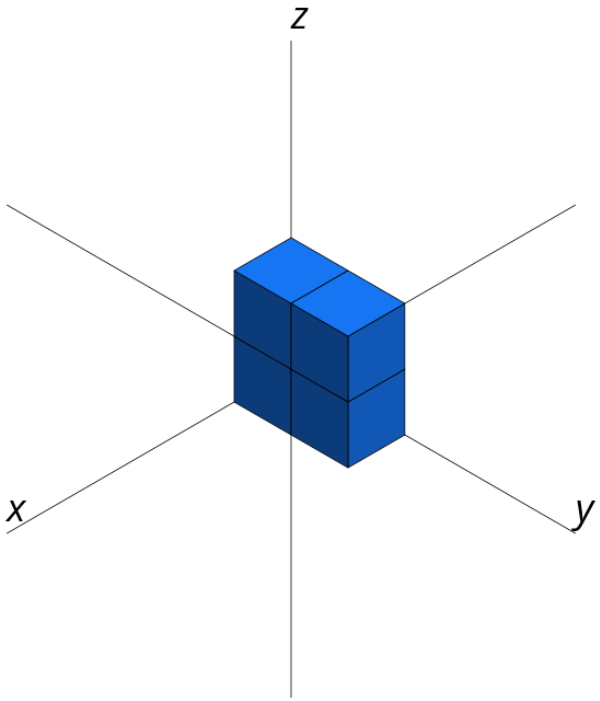
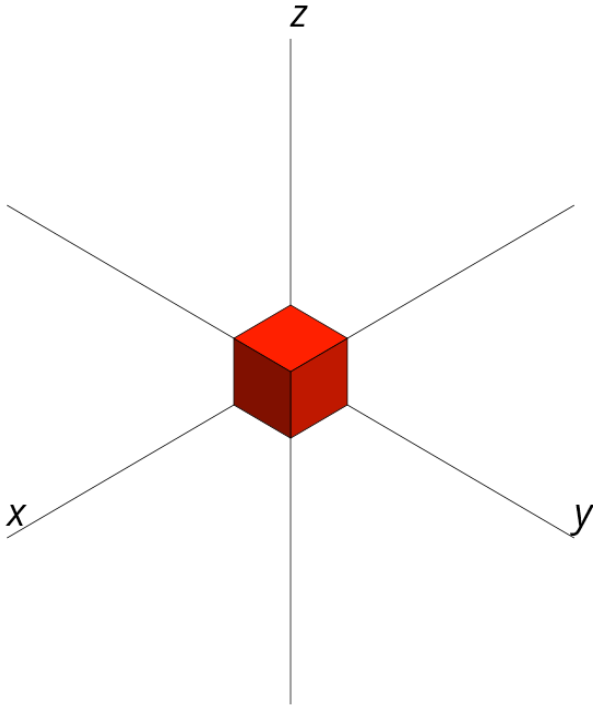
Output: (6) Scrap Value: $0.60	Explanation
	As shown in the pictures, a log with 6 in^3 cannot be cut by lumber 12 in^3. This shows that the cutting algorithm will scrap a log if the length of any lumber cuts are greater than its own.

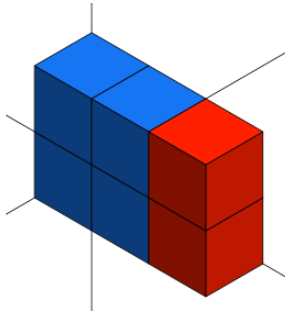
Test 4

Log: 2x3x1	Lumber: 2x3x1 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

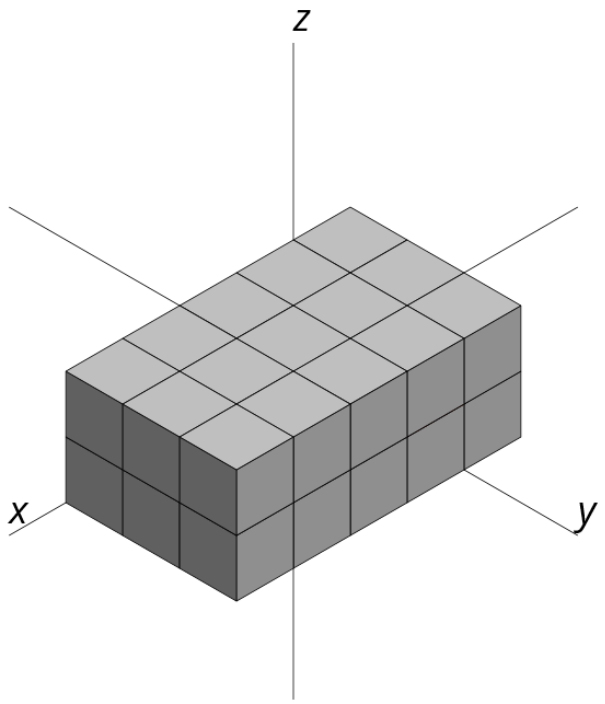
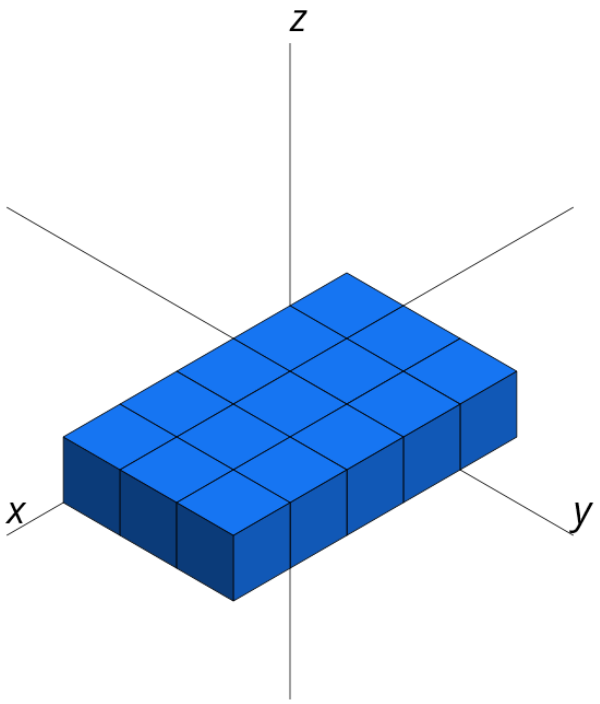
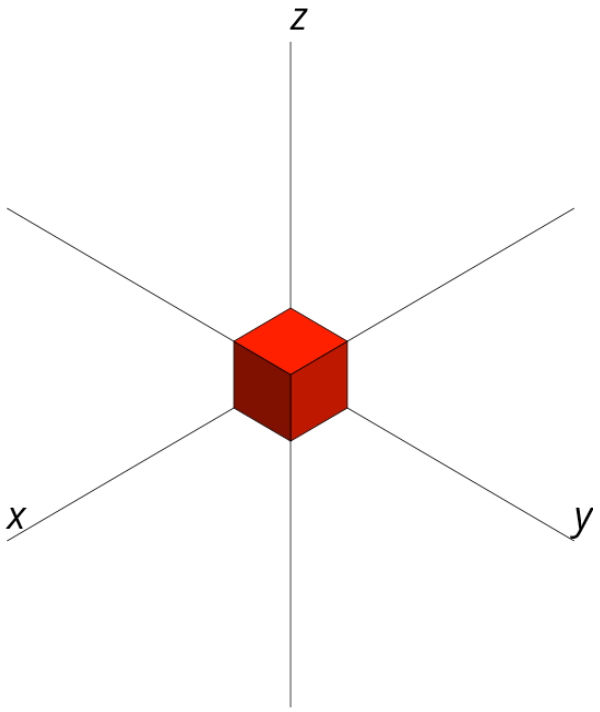
Output: (1) 2x3x1 Value: $1.00	Explanation
	As shown in the pictures, a log with 6 in^3 is perfectly cut by lumber size 6 in^3. This shows that the cutting algorithm will produce exactly one cut of lumber if the length matches the log length.

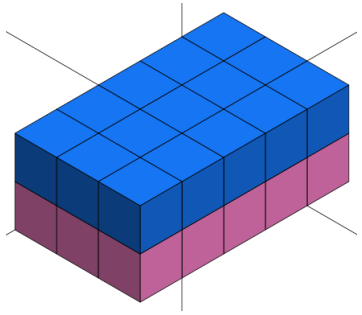
Test 5

Log: 2x3x1	Lumber: 2x2x1 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

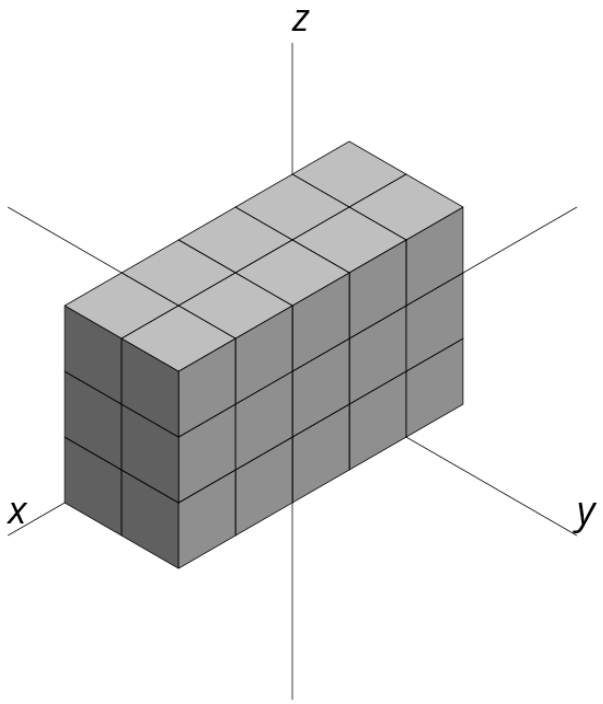
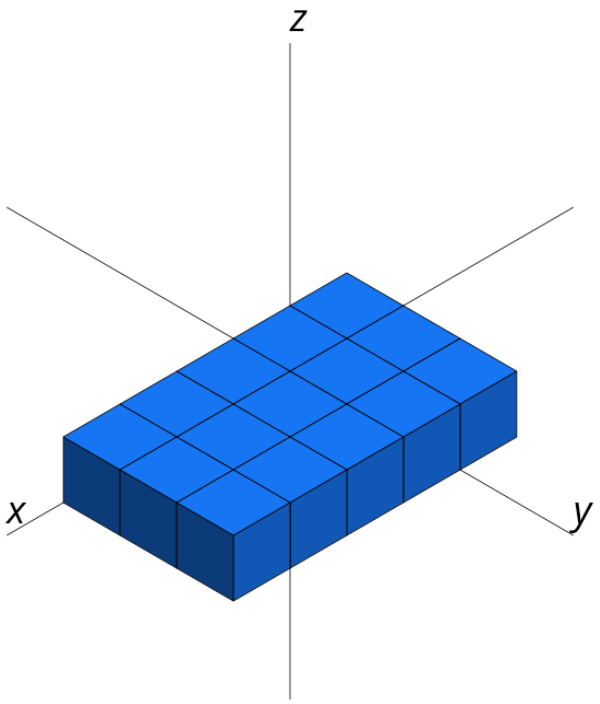
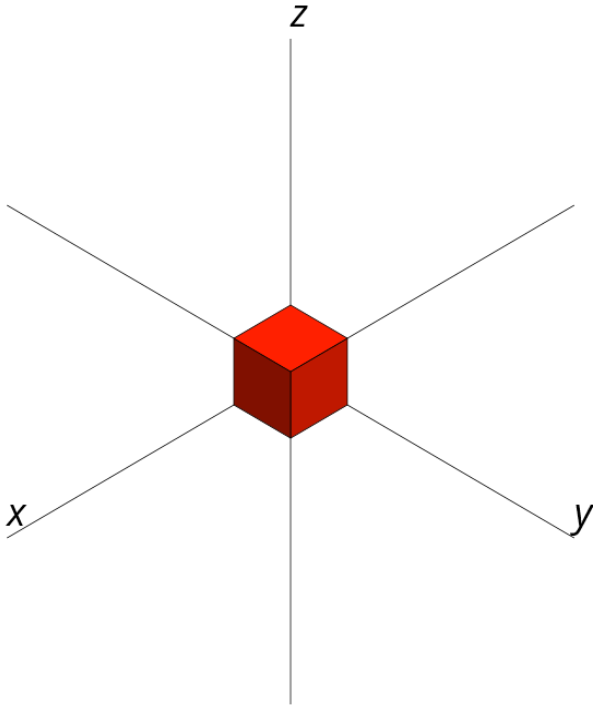
Output: (1) 2x2x1 (2) Scrap Value: $1.20	Explanation
	As shown in the pictures, a log with 6 in^3 is cut by lumber size 4 in^3. This shows that the cutting algorithm will produce output with a cut of lumber with the excess being scrapped if it cannot be cut into an appropriate lumber cut.

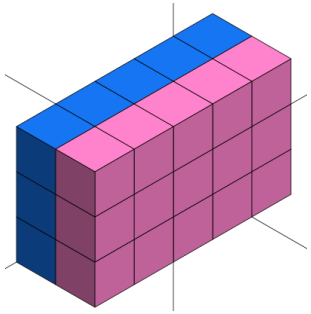
Test 6

Log: 2x3x5	Lumber: 1x3x5 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

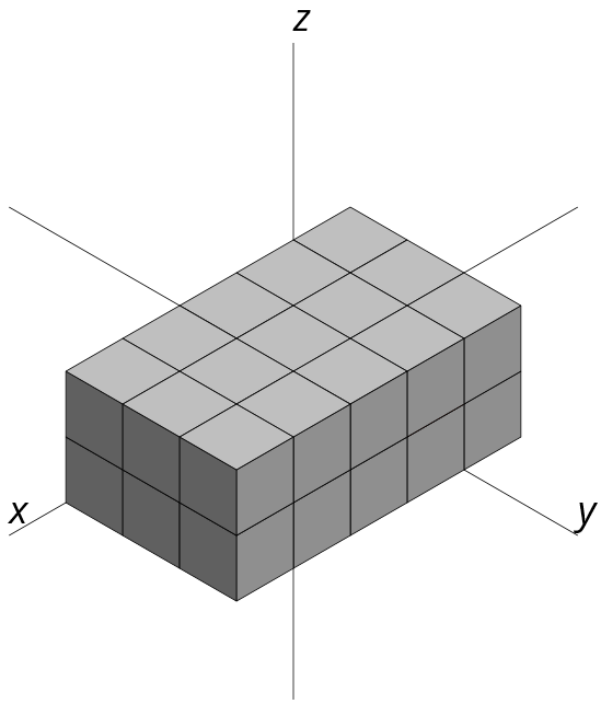
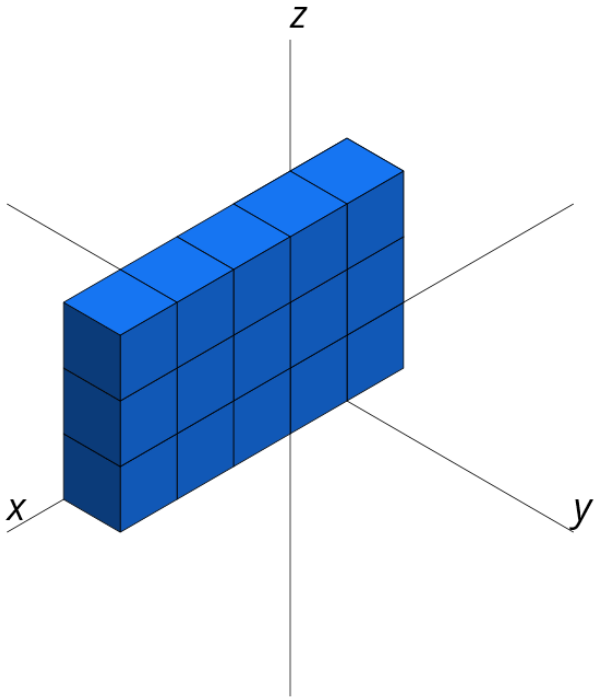
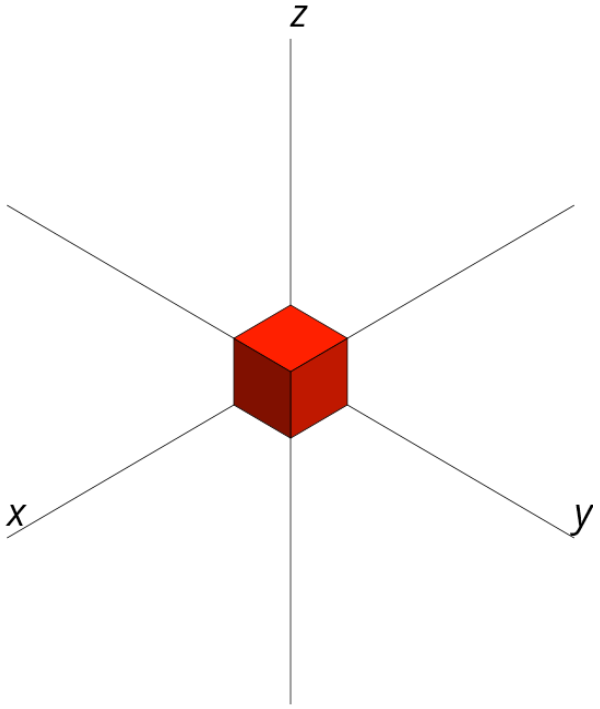
Output: (2) 1x3x5 Value: $2.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut by lumber size 15 in^3. This shows that the cutting algorithm can handle more than one lumber cut and serves as a base case for the next two cuts.

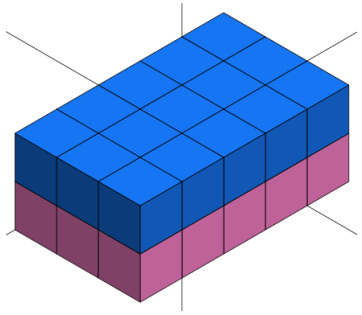
Test 7

Log: 3x2x5	Lumber: 1x3x5 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

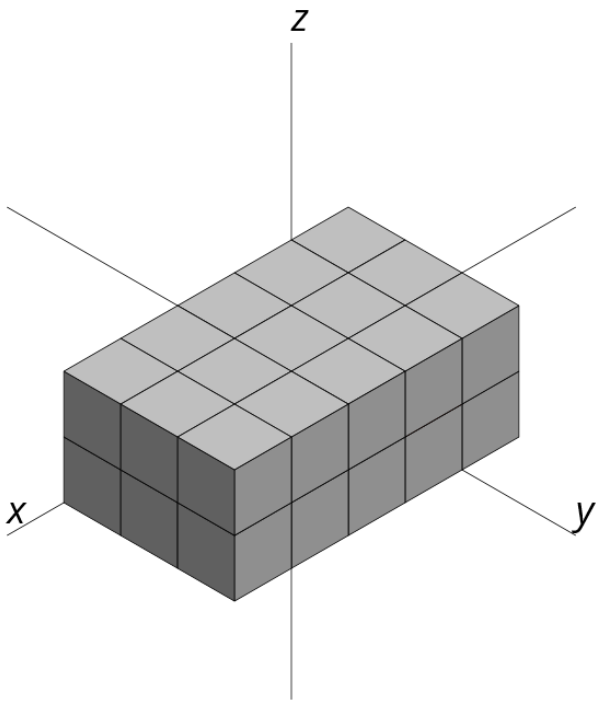
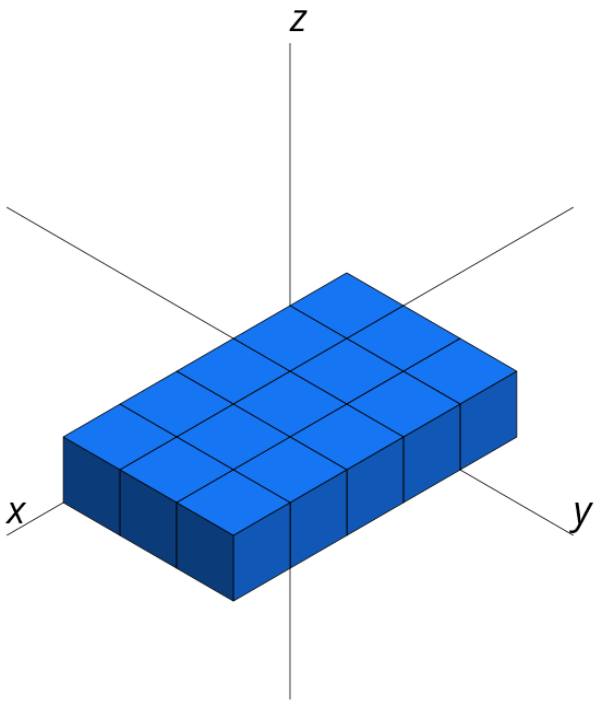
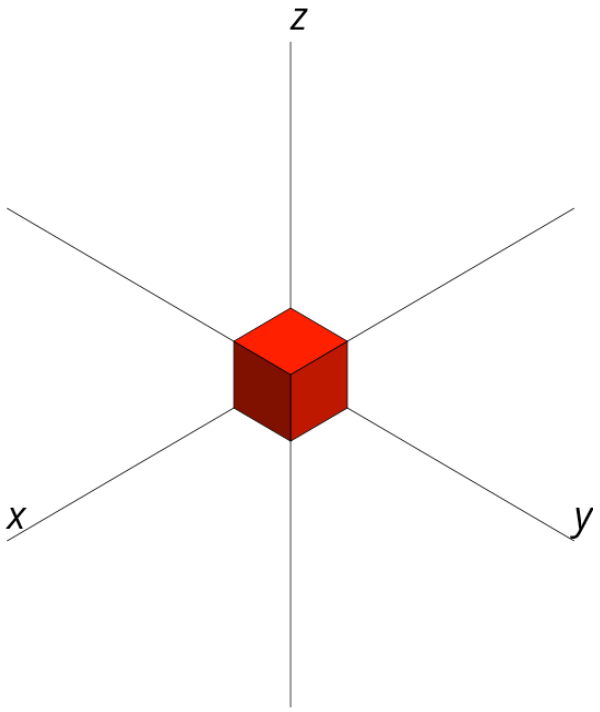
Output: (2) 1x3x5 Value: $2.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut by lumber size 15 in^3. This shows that the cutting algorithm will produce the same output if you rotate the log.

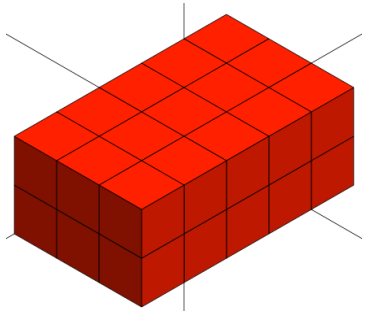
Test 8

Log: 2x3x5	Lumber: 3x1x5 Value: $1.00	Scrap: 1x1x1 Value: $0.10
		

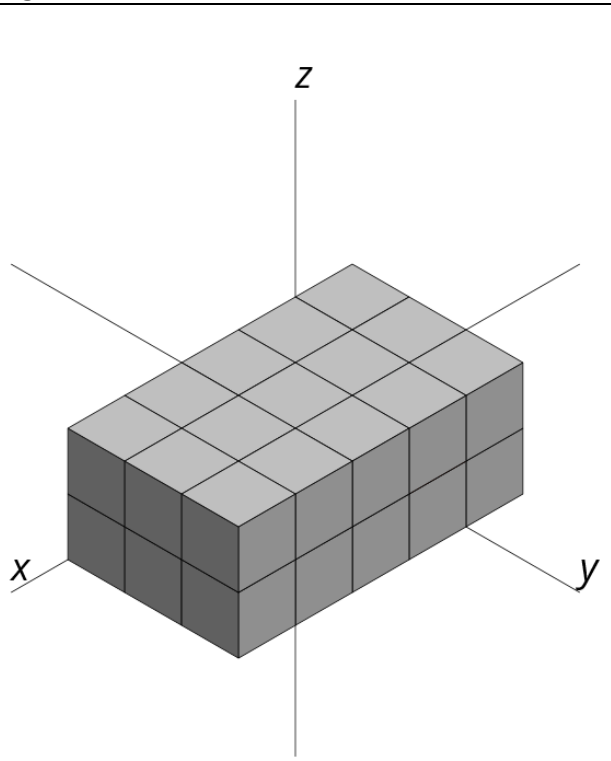
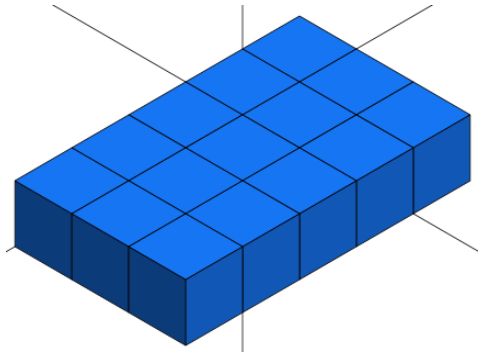
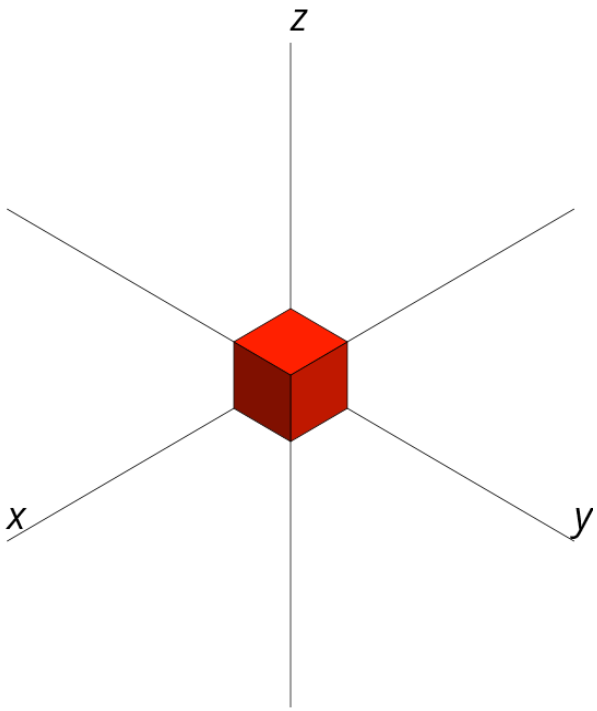
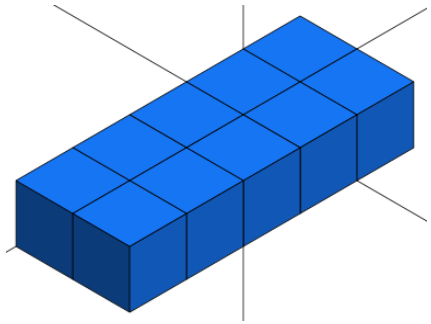
Output: (2) 1x3x5 Value: $2.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut by lumber size 15 in^3. This shows that the cutting algorithm will produce the same output if you rotate the lumber.

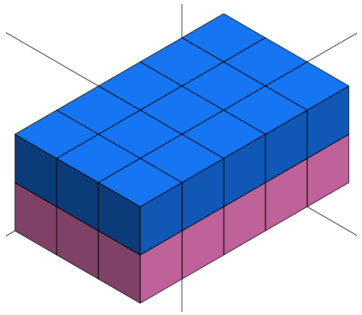
Test 9

Log: 2x3x5	Lumber: 3x1x5 Value: $1.00	Scrap: 1x1x1 Value: $1.00
		

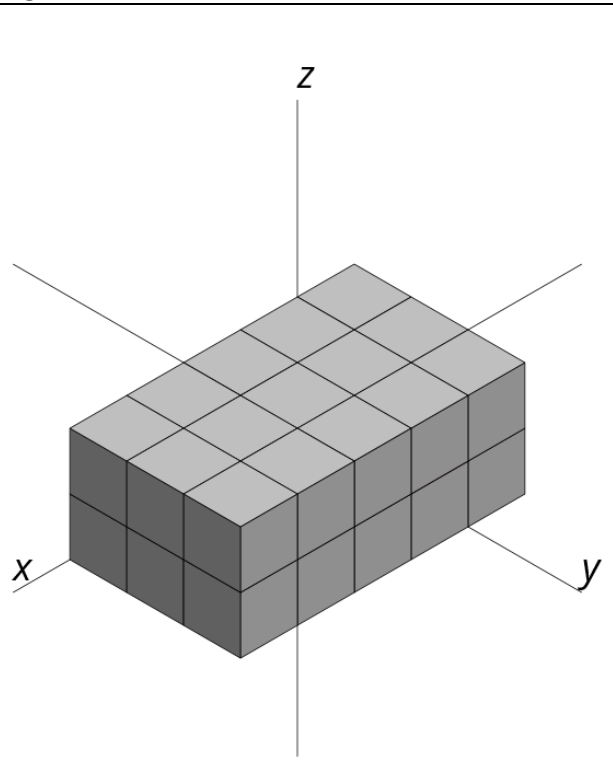
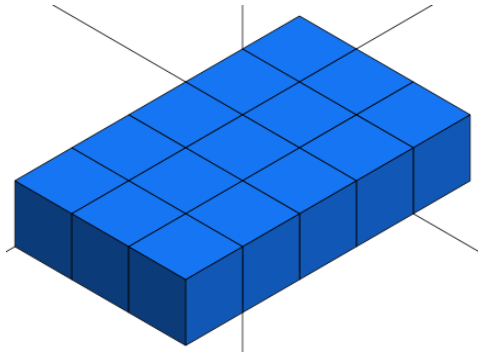
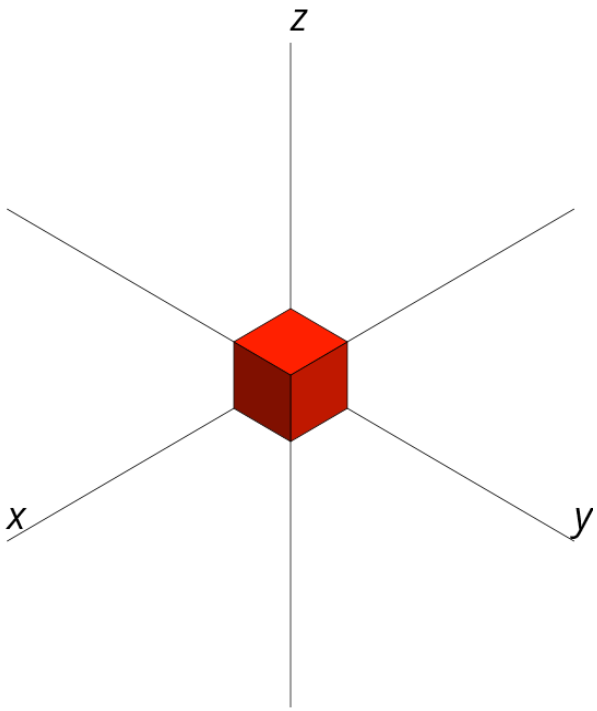
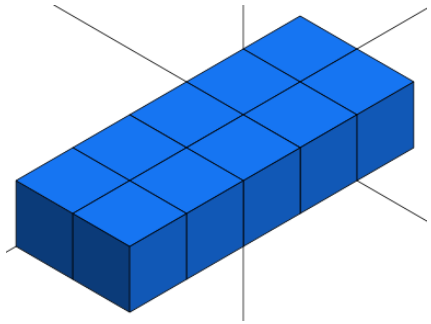
Output: (30) Scrap Value: $30.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut into scrap sized 1 in^3. This shows that the cutting algorithm will produce scrap if the value of the scrap is higher than any lumber cuts.

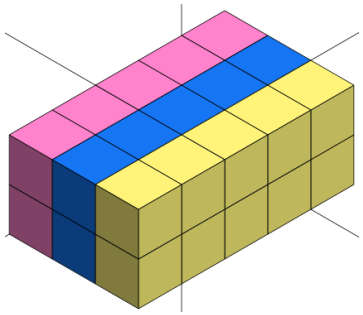
Test 10

Log: 2x3x5	Lumber 1: 1x3x5 Value: $3.00	Scrap: 1x1x1 Value: $0.10
		
	Lumber 2: 2x2x5 Value: $2.00	
		

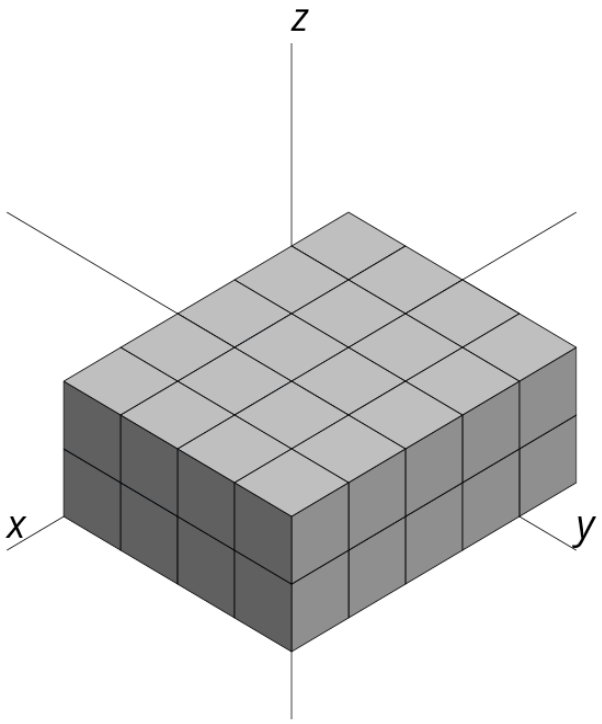
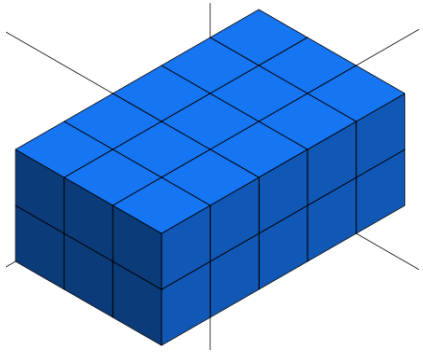
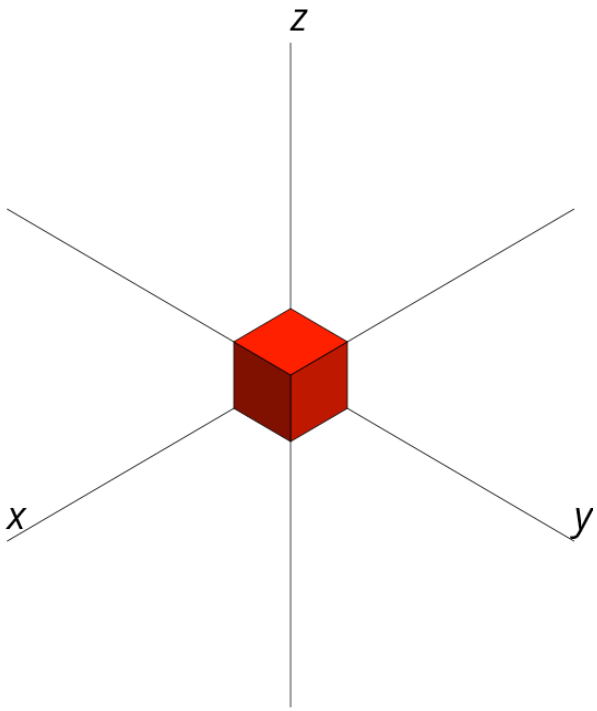
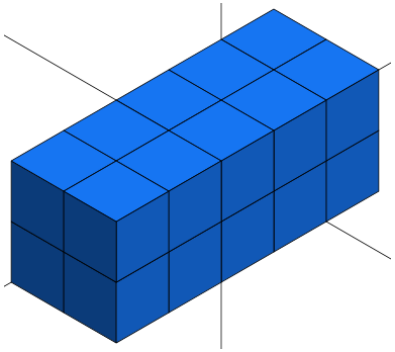
Output: (2) 1x3x5 Value: $6.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut by lumber size 15 in^3. This shows that the cutting algorithm will produce maximum output given two cuts of lumber.

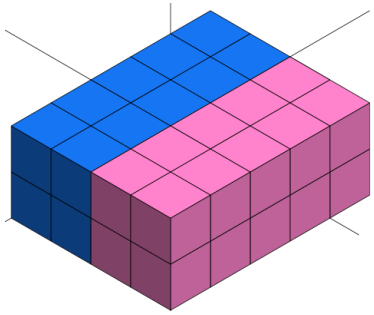
Test 11

Log: 2x3x5	Lumber 1: 1x3x5 Value: $2.00	Scrap: 1x1x1 Value: $0.10
		
	Lumber 2: 2x2x5 Value: $3.00	
		

Output: (3) 2x2x5 Value: $9.00	Explanation
	As shown in the pictures, a log with 30 in^3 is cut by lumber size 10 in^3. This shows that the cutting algorithm will produce maximum output given second cut of lumber is greater value even though it has less square inches than a larger cut.

Test 12

Log: 2x4x5	Lumber 1: 2x3x5 Value: $3.00	Scrap: 1x1x1 Value: $0.10
		
	Lumber 2: 2x2x5 Value: $2.00	
		

Output: (2) 2x2x5 Value: $4.00	Explanation
	As shown in the pictures, a log with 40 in^3 is cut by lumber size 20 in^3. This shows that the cutting algorithm will produce maximum output even though the first cut has higher square inches and value. This directly proves that the algorithm doesn't use the greedy method, but instead brute forces the log for the proper cuts.

Lumber Sold Test

Test	Specification	Test Case	Results
1	A:[exists] B:[DNE]	A=Hahira B=False	Invalid Input (B)
2	A:[DNE] B:[exists]	A=Atlanta B=True	Lumber bought/Shipped to Atlanta with price per mile
3	A:[exists] B:[exists]	A=Hahira B=True	Lumber bought/Shipped to Hahira with flat rate

Test	Purpose
1	Show that attempting to buy lumber out of stock will not be accepted.
2	Show that mailing lumber to non-flat rate location results in price per mile.
3	Show that mailing lumber to flat rate location results in flat rate.

Generic "Lumber Sold Test" Directions:

1. Run the system.
2. Upload log data & lumber data.
3. Perform cut logs command.
4. Select lumber pertaining to specific "Lumber Sold Test."
5. Input mailing address pertaining to specific "Lumber Sold Test."
6. Select purchase lumber.
7. Verify output matches expected results.
8. Perform steps 2-7 for each test in the "Input Lumber Data Test" set.
9. Exit the system.