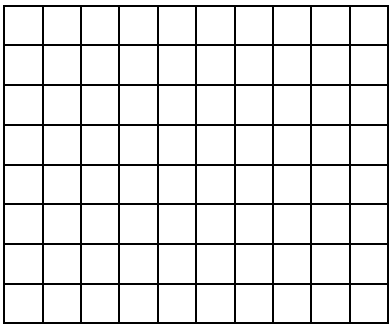


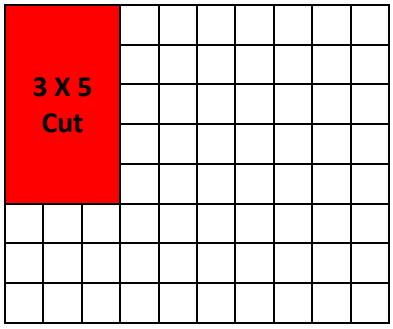
Let's assume you have a 10X8 log face like the one above and wish to cut the biggest cut of lumber possible, which happens to be a 3X5. Since height/width of lumber cuts are interchangeable this can be accomplished in two ways, we have to implement a check for both to ensure that the cut can be made:

First: IF [LumberWidth <= LogWidth && LumberHeight <= LogHeight] – The first check is pretty straightforward and ensures that you can make an appropriate cut.

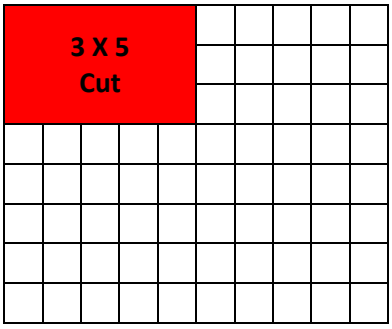
Second: IF [LumberHeight <= LogWidth && LumberWidth <= LogHeight] – The second check flips the height/width check to see if you can rotate the lumber and make an appropriate cut.

The resulting outcomes:

1

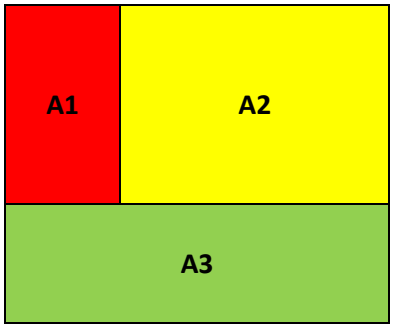


2

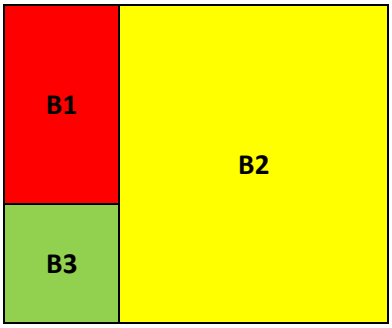


This splits our initial problem of 10X8 into two smaller sub problems based on how you decide to cut the out the lumber. Now, we must cut the remaining portions of the log as well. It is only logical that each remaining portion of the cuts be a rectangle as well in order to maximize possible cuts. This leaves us with two additional variations based on how the initial cut was made.

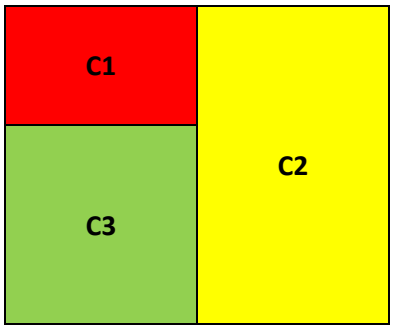
1A



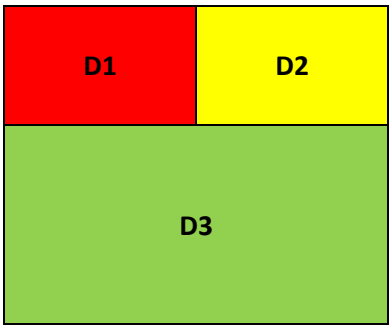
1B



2C



2D



There is no way of knowing which of these 4 cuts will be most optimal, so we must brute force until all variations are exhausted. This method is completed until no cuts can longer be removed, and the most efficient cuts are returned via the recursive method I created. Using the above illustrations, the algorithm becomes very easy to follow:

1A = A1 + A2 + A3
- A1 = [LumberWidth, LumberHeight]
- A2 = [LogWidth – LumberWidth, LumberHeight]
- A3 = [LogWidth, LogHeight – LumberHeight]

1B = B1 + B2 + B3
- B1 = [LumberWidth, LumberHeight]
- B2 = [LogWidth – LumberWidth, LogHeight]
- B3 = [LumberWidth, LogHeight – LumberHeight]

2C = C1 + C2 + C3
- C1 = [LumberHeight, LumberWidth]
- C2 = [LogWidth – LumberHeight, LogHeight]
- C3 = [LumberHeight, LogHeight – LumberWidth]

2D = D1 + D2 + D3
- D1 = [LumberHeight, LumberWidth]
- D2 = [LogWidth – LumberHeight, LumberWidth]
- D3 = [LogWidth, LogHeight – LumberWidth]