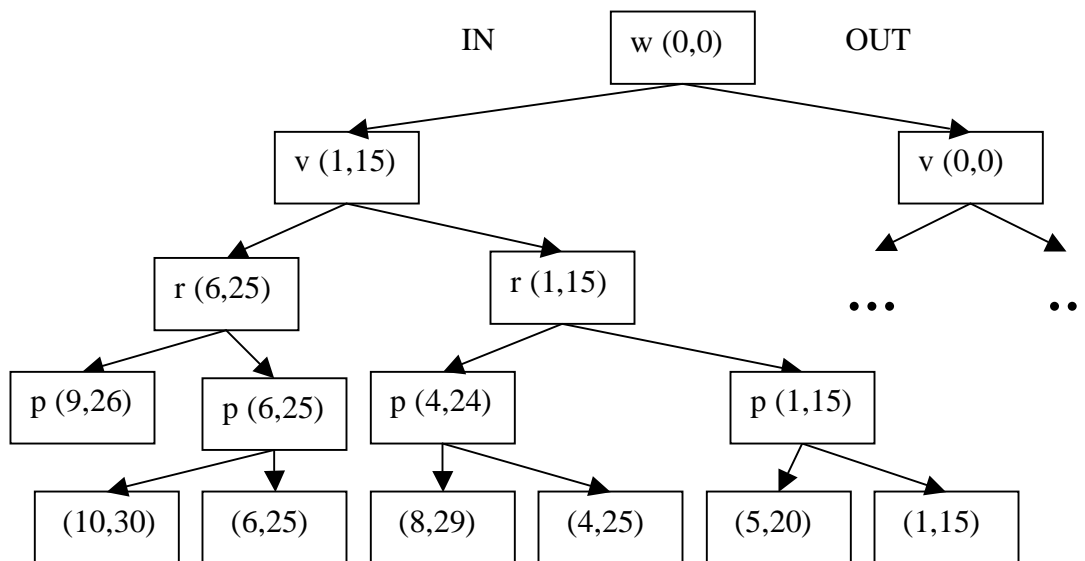


**6.00 Handout, Lecture 10**  
**(Not intended to make sense outside of lecture)**

Item	Size	Value
watch (w)	1	15
vase (v)	5	10
radio (r)	3	9
picture (p)	4	5



**Branch-and-bound**

1) First we need a way of covering the feasible region by splitting into sub-regions (remember divide and conquer). This is called **branching**, since the procedure is repeated recursively to each of the sub-regions and all produced sub-regions form a tree.

2) Secondly, we need a method of **bounding**, which is a fast way of finding upper and lower bounds for the optimal solution within a feasible sub-region.

What do we mean by this? For the knapsack problem, a lower bound is a value that is guaranteed to be at least as high as the value at the end of the path. An upper bound is a value that is guaranteed to no greater than the value at the end of the path.

The idea behind branch and bound is the observation that for a maximization task if the *upper* bound for a sub-region *A* from the search tree is less than the *lower* bound for any other (previously examined) sub-region *B*, then *A* may be safely discarded from the search.

We can backtrack if we know the best possible solution in the current sub-tree is worse than current best solution obtained so far. This step is called **pruning**.