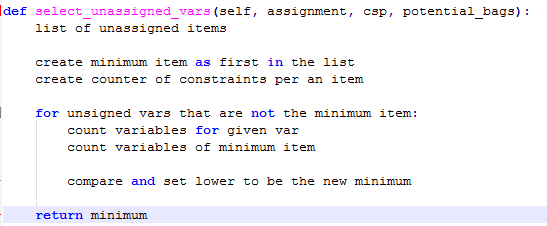
Constraint Satisfaction Problems (CSPs) are problems with a number of objects that must be placed in a state which satisfies the given constraints of the problem. We created a function to solve a bag backing problem given a number of objects and their weights and a number of bags with a maximum weight capacity. Our program was able to efficiently fill the bags to the given constraints. We utilized Python version 2.7 to create this program.

Our program uses backtracking search and forward search in order to achieve a higher level of efficiency. This allows us to prune unnecessary nodes in order to increase our efficiency and minimize search time. We utilized a Minimum Remaining Value based heuristic in our program. This heuristic chooses the most constrained variable, as it has the fewest legal values available. Given a tie we used a degree heuristic to choose the most constraining variable or the variable that would cause the most constraints on remaining variables. Our heuristic was based off the given pseudocode.



During backtracking we also utilized a Least Constraining Value heuristic to create a list of values ordered from the least to most constraining. This list gives the values that allow the fewest values possible for the remaining variables. This means that for a given value the least constraining variable would allow for other variables in the problem to have the most possible values which gives us the most flexibility for future solutions. The Least Constraining Value heuristic is shown in the below pseudocode:

