Assignment 2 Write up

Much like the previous assignment, I have focused on the speed aspect of the algorithm. I wanted to produce a model that outputs the model rather instantly. Therefore, I created an algorithm that closely mimics the model I have created for the knapsack problem. I first sorted the list of items from the heaviest to the lightest. Then, I fit in the items to the bin, but when the item is too big to fit into the resting capacity of the bin, I opened a new one.

The overall structure of the algorithm is the same as the one that was provided by Professor Wilck. In fact, the code provided was the simplest algorithm I could find, that goes through the least iteration as well. As Professor Wilck described in class, we first create lower bound *lowbound* on the number of bins, then, create the minimum number of bins through a variable called *binlist.* Then we create the volume cap of each bin as *volcap*. Then we iterate through the list of items and the bins to find if the volume of an item is less than the *volcap* of the bin. If it is, we append the item and update the *volcap* by subtracting the item volume out. However, if the item volume is too big, we break the loop and open a new bin. This process continues until all items are assigned.

However, with a minor differences. I sorted the list of items from heaviest to the lightest. I did this to create a *first fit decreasing* algorithm, which does suffer from achieving high accuracy, but it is indeed a very fast algorithm for this type of problem. However, as I have mentioned before, I wanted to create a fast algorithm.

* The first fit decreasing algorithm, much like its name, it finds and places the item to a first bin that could fit the item. The order of items being stored starts from the biggest items and ends with the smallest item.