**WestonAssignment4.cpp Purpose and Design**

WestonAssignment4 uses dynamic programming to reduce redundant calculations by breaking the shipping container optimization problem into smaller pieces, solving increasingly larger cases, and storing solutions in a data structure. Optimal shipping container lengths are created and maintained as OptimalContainer objects. OptimalContainer objects have length, value, and “components” attributes. “Components” is a vector of sorted OptimalContainer objects representing the optimized “sub-containers” that comprise the larger OptimalContainer.

The program maps integers (representing container length) to OptimalContainer objects in a hash table. The algorithm to determine a length’s OptimalContainer iteratively compares that length’s bid price to other valid combinations of previously optimized OptimalContainers. The values of different combinations of OptimalContainers are efficiently accessed via the hash table. In this way, the algorithm greatly reduces the number of comparisons and never performs redundant calculations.