COMPUTER SCIENCE & ENGINEERING

CSE2046 – ANALYSIS OF ALGORITHMS

HOMEWORK #2 – REPORT

Abbas Göktuğ YILMAZ – 150115061

Enes Garip – 150116034

Veysi Öz – 150116005

Our Algorithm

First of all, when we started the homework, we misunderstood a part of the homework. Our algorithm solves all the knapsacks in an input file individually, and returns the value of the highest value knapsack. Here's where we got it wrong. While solving the knapsacks, we realized that the same elements must not exceed the capacity in all knapsacks when our outputs did not pass through the verifier. As a result, our algorithm is not a "Multiconstraint Knapsack", it is normal knapsack algorithm.

Our Normal Knapsack Algorithm

- The value of each item is divided by the weight of that item. Thus, the value of the unit weight of that item is found.
- The item with the highest value we found in the previous step is added to the knapsack.
- We reset the value of the item we received so that it does not return the same element when finding the highest valued item again.
- In this way, we take the items and sum their weights. The algorithm stops when the sum reaches the capacity of the knapsack.

The Memory Used by Code

Before starting our algorithm, we used arraylists to Access the data in an organized manner. So we kept:

- the values in vValues(arraylist),
- the Knapsack Capacities in knapsackCapacities(arraylist),
- the weights in weights(arraylist).

The Complexity of Code

The place where the complexity increases the most in our code is between lines 132-147(while loop). The complexity of this line spacing, that is, the whole code, is ***O(n)***.