Homework 18: Experiment

CSC-351

December 5th, 2023

Experiments and Report for the Exact Change Problem

Combo for experiment: *Many coins. Coin denominations are small. Change is relatively large.*

I chose to compare my Greedy and Recursive Dynamic Programming algorithms. Both of my Dynamic Programming algorithms are not working properly, so I will write a bit about both my results and what the program should actually do.

Based on the results, my Greedy algorithm was more accurate than my Recursive Dynamic Programming algorithm. My Greedy algorithm found the solution everytime, this is because my coin range included “1” (the coins array generally had multiple “1” values as well which almost guaranteed finding the solution) and the algorithm was working properly. RecDP on the other hand got the correct solution about half of the time, this is because the algorithm was not working properly and I couldn’t figure out how to fix it. Theoretically RecDP should’ve found the solution everytime, since the coin pool was of limited size. It also seems like (from the choices made) that the Greedy was also more efficient in terms of least items selected. While RecDP was way less efficient when picking values as shown by the massive amounts of chosen values.

Both algorithms had the same count and time, which I chould’ve improved Greedy by adding a “break” condition to it’s loop if the solution was found. So based on my actual code, the count seems on point. Greedy could’ve had a better count and time but RecDP won’t get any better due to the nature of having to traverse the entire memoization table.

Results below ↓ (I only ran the algorithm 5 times keeping all variables the same for this experiment)

Link to spreadsheet: Algorithms Homework 18

Algorithm	Choices Made	Change Required	Change Obtained	Time	Operations/Count	Num Coins	Change Range	Coin Range
Greedy	30,30,30,30,12	132	132	951	17201	2000	60 - 160	1 - 30
Greedy	30,30,30,30,30,1	151	151	961	17201	2000	60 - 160	1 - 30
Greedy	30,30,30,23	113	113	859	17201	2000	60 - 160	1 - 30
Greedy	30,30,30,30,11	131	131	1097	17201	2000	60 - 160	1 - 30
Greedy	30,30,30,15	105	105	996	17201	2000	60 - 160	

RecDP	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	132	132	951	17201	2000	60-160	1 - 30

	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1							
RecDP	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	151	150	961	17201	2000	60 - 160	1 - 30

	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1							
RecDP	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	113	113	859	17201	2000	60 - 160	1 - 30

	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1							
RecDP	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	131	130	1097	17201	2000	60 - 160	1 - 30

	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1							
RecDP	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	105	105	996	17201	2000	60 - 160	1 - 30

	1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1							