THE UNIVERSITY OF CHICAGO CAAP, COMPUTER SCIENCE

CAAP AP CS Assignment 1

Dawei Huang

July 17, 2018

1 WHAT IS A GREEDY ALGORITHM?

A greedy algorithm is a type of algorithm that intends to make optimal decisions at every step of the algorithm without taking into consideration what might or might not be the optimal decision in the context of the entire algorithm. Therefore it is entirely possible that taking each step optimally may not yield a process that is optimal in the context of the entire algorithm. Generally speaking, optimal step-taking should lead to an optimal general algorithm. In the context of our change algorithm, at every step, we make the optimal decision to choose a coin whose value is closest to but less than the amount of change we have left in question. As it turns out, this process of determining the least amount of coins in a specified amount of change is a generally optimal algorithm.

2 HOW ELSE COULD YOU SOLVE THE ABOVE PROBLEM?

Besides using the greedy algorithm to solve the problem, I can instead directly choose how many coins could fit into the amount of change we have in question starting with higher denomination coins. For example, given that we have X value of change, we find the maximum number of quarters whose sum is under that change value. Then we take the difference of x and the maximum allowed value of the quarters we have. We then use this process to move on to dimes, then nickels, and then finally to pennies.

3 What other problems are greedy algorithms successful for?

Greedy algorithms are successful in pretty much any algorithm that can be simplified and broken down into a number of similar and pseudo-repetitive steps. An example of this would be all different kinds of sort algorithms that would take a set of numbers in random numbers and order them based on their unique algorithms. We can also use greedy algorithms to parse through trees and find its length, and etc.