3^{rd} Assignment
CS430 Introduction to Algorithm, Fall 2019

Instructor: Edward Chlebus
TA: Haohua Du, Yidan Li

November 25, 2019

This homework is due at 11:59pm on Dec 6, 2019

Assignment Instruction

- Team work is allowed, max 4x students per team.
- ONLY team leader submits the PDF version of the assignment to the Blackboard. You also HAVE TO include all the team members' full name and A-number in the first page of the submission.
- Late submissions won't be accepted.
- All solutions should be explained.
- Extra credits would be given for nonstandard original solutions.

Problem 1 (20pts)

There are alternatives to CLRS3's greedy choice of the earliest finishing time the activity selection problem in section 16.1. For each of the following alternatives, prove or disprove that it constructs an optimal schedule (assume ties are broken arbitrarily):

1. Choose the job that starts last.
2. Choose the job that conflicts with the fewest other jobs.
3. Choose the job of longest duration.

Problem 2 (20pts)

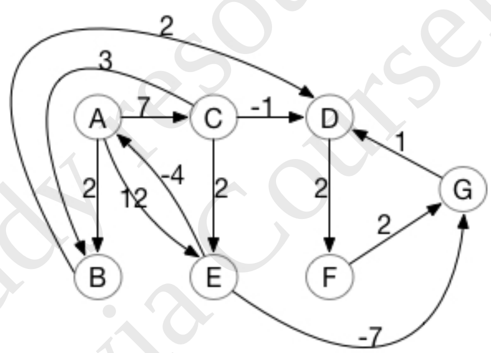
A country has n different types of coins with denominations $1 = c_1 < c_2 < \cdots < c_n$. You want to give change using the fewest number of coins. The greedy algorithm for giving change for an amount d subtracts the largest denomination coin from d and repeats recursively on the amount still needed.

1. We have $c_1 = 1, c_2 = 5, c_3 = 10, c_4 = 25, c_5 = 50$, and $c_6 = 100$. Does the greedy algorithm always give the change with the fewest coins? Either prove that it does, or provide a counterexample.

2. Suppose the denominations are consecutive powers of an integer $b \geq 2$; that is $c_1 = 1, c_2 = b, c_3 = b^2$, and so on. Does the greedy algorithm always give the change with the fewest coins? Either prove that it does, or provide a counterexample.

Problem 3 (20pts)

Consider the following directed, weighted graph:



1. Show your steps of using Dijkstra's algorithm for finding the shortest path in the table below. Cross out old values and write in new ones, from left to right within each cell, as the algorithm proceeds. Also list the vertices in the order which you marked them known.

2. Dijkstra's algorithm found the wrong path to some of the vertices. For just the vertices where the wrong path was computed, indicate both the path that was computed and the correct path.

Vertex	Known	Distance	Path
A			
B			
C			
D			
E			
F			
G			

Problem 4 (20pts)

Consider a flow network with non-negative edge capacities. Show that there always exists a maximum flow f in which either $f(u, v) = 0$ or $f(v, u) = 0$ for any pair of vertices.