

COR-IS1702 – Computational Thinking

Academic Year 2019-2020 Term 2

**Personal Project**

**Section No.: G4**

**Prepared By:**

Bui Phuong Thao

**1. Question 2**

Since the preferences may be in any order, my first idea is to create a helper function to convert the original preference list and the solution list into dictionaries to retrieve the value.

My approach to this question is: **Find one pair that is unstable**

Steps:

* Go through every pair in the solution
* Find the man’s rank in the woman’s preference list
* If her current partner’s rank number is larger than 1:
  + Find the woman’s list of preferred man from the highest ranked to her current partner
  + Loop through the list from the highest ranked man (A)
  + Find that man (A)’s preference list.
  + If the woman’s rank number in A’s preference list is smaller than A’s current partner (meaning A prefers the woman to his current partner), return False
  + Repeat that until the end of the list
* Else move to the next pair
* If no pair return False, return True

Complexity: O(n2)

* Convert lists to dictionaries: 2n iterations -> O(n)
* For loop: n iterations: in each iteration, there is another for loop with a maximum of n iterations

**2. Question 3:**

I adopted the Greedy Algorithm for this question, and got the reference from Lab 4’s Debrief Greedy Algorithm. I also used 3 helper functions in the utility file

My approach to this question is: **Find one Twitter’s user with highest value (taking into consideration his/her followers) and from that one user, find the next user with highest value. Repeat until the cost exceeds the budget**

Steps:

* Initialize total cost = 0, initialize an empty list that will contain the answer
* While cost < budget:
  + Create a temp list that contains all possible users to be chosen along with their values
    - Possible users to be chosen are the ones that are not in already in the answer and their costs are lower than their values
    - The value of a user is his/her value plus his/her followers’ values
  + If this list is empty (there is no possible users to be chosen), return the answer list
  + Else, Sort this temp list by the users’ value in descending order
    - Get the user with the highest value and append him/her into the answer list
    - Get that user’s followers
    - Update the followers list by removing followers who followed that user
    - Get the total cost of the users in ans and compare it to budget.

Complexity: O(n2 logn)

* Maximum iterations in the while loop: n (n followers)
* Inside an iteration in while loop:
  + A for loop with maximum: n iterations
  + An if-else statement: ▪ Python sort(): O(n logn) ▪ For loop: n iterations

**Appendices**

False

False

True

b’s rank in c’s pref < c’s partner’s

n, pref, sol

First pair [a, b]

Find a’s rank number in b’s preference

a’s rank > 1

Get b’s preference list

Initialize i = 1

Get the man with rank i in b’s pref (c)

Get c’s preference list

Return False

End

Increment i by 1

True

i < a’s rank

No pair left?

True

False

Next pair

False

Return True

End

True

Appendix A: Q2’s Flowchart

Appendix B: Q3’s Flowchart

False

True

False

False

True

False

True

flags, p, v

Initialize cost = 0

ans = []

cost < budget

Initialize user i = 0

i < len(f)

i not in ans and c[i] < v[i]

Get value of user i

Append to temp [i, value]

Initialize temp = []

len(temp) > 0

Sort temp

Get the 1st user in temp and append to ans

Get the 1st user’s followers

j < len(f)

Remove followers that followed 1st user in user j’s followers

Return ans except last element

End

Increment i by 1

Initialize user j = 0

Get cost

Increment j by 1

Return ans

End

True

Get cost

Initialize user j = 0

**Reference**

1. COR-IS1702 CT Lab 4 Debrief: <https://www.youtube.com/watch?v=12053Wc8umo&feature=youtu.be>
2. Project1 (v1.0)/q3/p1q3\_utility