Problem-1

i) inputs: people, individuals output: a list of the matches

ii) Let's say that we have a group of people and their universities as input. We want to match these people

with their own different universities. At the end, we will have a matched list as output.

Problem-2

i)

Initialize all men and women to free

while there is a single free man m who has a woman w to propose to

w := the first woman who m has not yet proposed to her and in m's list

if w is free then

m and w become engaged

else some pair such that m'and w already exists

if w prefers m to m' then

m' becomes free

m and w become engaged

else

m'and w remain engaged

end if

end if

repeat

ii) Let's say that # of men = # of women = n then we can say that 2\*n\*n then complexity becomes O(n^2)

Problem-3

# men = {A, B, C, D}

# women = {J, K, L, M}

# Initialize all men and women to free (in here I use True for being free)

# My inputs are dictionaries (men, women) in here and

# number of men = number of women = n = 4 in this example

men = {'A': True, 'B': True, 'C': True, 'D': True}

women = {'J': True, 'K': True, 'L': True, 'M': True}

# At the end, I want such a match

match = {'A': 'K', 'B': 'L', 'C': 'J', 'D': 'M'}

# key, value

for man, freeM in men.items():

if freeM == True:

for woman, freeW in women.items():

if freeW == True:

if(match.get(man) == woman):

print( man, ' and ', woman, ' become engaged.' )

freeM == False

freeW == False