

HW |

Problem 1

The statement is false. Consider the example w/ 2 interns, i_1 & i_2 , and 2 companies, C_1 & C_2 .
 C_1 preference list

1. i_1

2. i_2

C_2 preference list

1. i_2

2. i_1

i_1 preference list

1. C_2

2. C_1

i_2 preference list

1. C_1

2. C_2

It is impossible for there to be a stable matching containing a pair (c, i) s.t. c is ranked first on the preference list of i and i is ranked first on the preference list of c . Thus the statement is false.

Problem 2

If there is a stable matching of workers to jobs, that means every worker w is linked to a job j and there are no instabilities. Since there are the same amount of full-time jobs, part-time jobs, hard workers, and lazy workers, if a full-time job is linked with a lazy worker, then a hard-worker was linked with a part-time job. This is an instability since full-time jobs and hard-workers prefer each other, so this cannot happen in a stable matching. So every hard worker gets a full-time job in a stable matching.

Problem 3

modified (a) initially all $c \in C$ and $i \in I$ are free
self-shaply while $\exists c$ who is available and hasn't
offered to every $i \in I$

do choose such a company c .

let i be the highest ranked in
 c 's preference list to whom m has
not yet made an offer.

if there exist multiple i with the
same preference ranking, make an
offer to a random one.

if i is free

then (c, i) become linked.

else i is currently to c'

if i prefers c' to c // c & c' are not tied

then c remains free // c & c' are tied or i prefers c to c'

else i prefers c to c'

(c, i) become linked

c' becomes free

return the set S of linked pairs

(b) C_1 preference list C_2 preference list

1. i_1, i_2

1. i_2, i_1

i_1 preference list

i_2 preference list

1. C_1

1. C_1

2. C_2

2. C_2

all perfect matchings have a weak instability
 $(C_1, i_1), (C_2, i_2)$ & $(C_1, i_2), (C_2, i_1)$

Problem 4

Suppose c_i and i_j are matched in S , but c_i is not the worst valid partner for i_j . Then there exists a stable matching S' where c_k and i_j are matched, though i_j prefers c_i over c_k ; and c_i is matched with i_k . However, according to S , c_i prefers i_j since Gale-Shapley returns the company-optimal matching. This indicates an instability, thus S' is not a stable matching. Hence, there is no valid matching where i_j has a worse valid match. Thus, Gale-Shapley returns the intern-perimal matching.