CS 70 Discrete Mathematics and Probability Theory
Fall 2024 Rao, Hug

HW 02

Due: Saturday, 9/13, 4:00 PM
Grace period until Saturday, 9/13, 6:00 PM

Sundry

Before you start writing your final homework submission, state briefly how you worked on it. Who else did you work with? List names and email addresses. (In case of homework party, you can just describe the group.)

1 Airport

Note 3

Suppose that there are $2n+1$ airports, where n is a positive integer. The distances between any two airports are all different. For each airport, exactly one airplane departs from it and is destined for the closest airport. Prove by induction that there is an airport which has no airplanes destined for it.

2 Grid Induction

Note 3

Pacman is walking on an infinite 2D grid. He starts at some location $(i, j) \in \mathbb{N}^2$ in the first quadrant, and is constrained to stay in the first quadrant (say, by walls along the x and y axes).

Every second he does one of the following (if possible):

(i) Walk one step down, to $(i, j-1)$.

(ii) Walk one step left, to $(i-1, j)$.

For example, if he is at $(5,0)$, his only option is to walk left to $(4,0)$; if Pacman is instead at $(3,2)$, he could walk either to $(2,2)$ or $(3,1)$.

Prove by induction that no matter how he walks, he will always reach $(0,0)$ in finite time.

(*Hint*: Try starting Pacman at a few small points like $(2,1)$ and looking all the different paths he could take to reach $(0,0)$. Do you notice a pattern in the number of steps he takes? Try to use this to strengthen the inductive hypothesis.)

3 Universal Preference

Note 4

Suppose that preferences in a stable matching instance are universal: all n jobs share the preferences $C_1 > C_2 > \cdots > C_n$ and all candidates share the preferences $J_1 > J_2 > \cdots > J_n$.

(a) What pairing do we get from running the algorithm with jobs proposing? Can you prove this happens for all n?

(b) What pairing do we get from running the algorithm with candidates proposing?

(c) What does this tell us about the number of stable pairings?

4 Pairing Up

Note 4

Prove that for every even $n \geq 2$, there exists an instance of the stable matching problem with n jobs and n candidates such that the instance has at least $2^{n/2}$ distinct stable matchings.

5 Optimal Candidates

Note 4

In the notes, we proved that the propose-and-reject algorithm always outputs the job-optimal pairing. However, we never explicitly showed why it is guaranteed that putting every job with its optimal candidate results in a pairing at all. Prove by contradiction that no two jobs can have the same optimal candidate. (Note: your proof should not rely on the fact that the propose-and-reject algorithm outputs the job-optimal pairing.)