

Due: Friday 09/11 at 10:00 PM
Grace period until Friday 09/11 at 11:59 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 Induction

Prove the following using induction:

- (a) For all natural numbers $n > 2$, $2^n > 2n + 1$.
- (b) For all positive integers n , $1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$.
- (c) For all positive natural numbers n , $\frac{5}{4} \cdot 8^n + 3^{3n-1}$ is divisible by 19.

2 Negative pacman returns

Pacman has had a bit of a wild night, and wakes up feeling a bit under the weather. He starts at some location $(i, j) \in \mathbb{N}^2$ in the third quadrant, and is constrained walk on the infinite 2D grid and stay in the third quadrant (say, by walls along the negative x and negative y axes). Every second he does one of the following (if possible):

- (i) Walk one step up, to $(i, j + 1)$.
- (ii) Walk one step right, to $(i + 1, j)$.

For example, if he is at $(-5, 0)$, his only option is to walk right 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?)

3 Losing Marbles

Two EECS70 GSIs have inexplicably run out of research topics to pursue, papers to read, or homeworks to create, and so they decide to play an incredibly boring game. (This is EECS after all.)

In the game, there is an urn that contains some number of red marbles (R), green marbles (G), and blue marbles (B). There is also an infinite supply of marbles outside the urn.

When it is a player's turn, the player may either:

- (i) Remove one red marble from the urn, and add 3 green marbles.
- (ii) Remove two green marbles from the urn, and add 7 blue marbles.
- (iii) Remove one blue marble from the urn.

These are the only legal moves. The last player that can make a legal move wins. We play optimally, of course – meaning we always play one of the best possible legal moves.

- (a) If the urn contains (R, G, B) red, green, and blue marbles initially, then determine the conditions on R, G, B for the first player to win the game. Prove it. In this case, does it matter what strategy the players use?

Hint: Assign each marble a weight, and argue that at every step, the combined weight will go down by exactly 1.

- (b) Prove by induction that, if the urn initially contains a finite number of marbles at the start of the game, then the game will end after a finite number of moves.

4 Nothing Can Be Better Than Something

In the stable matching problem, suppose that some jobs and candidates have hard requirements and might not be able to just settle for anything. In other words, in addition to the preference orderings they have, they prefer being unmatched to being matched with some of the lower-ranked entities (in their own preference list). We will use the term entity to refer to a candidate/job. A matching could ultimately have to be partial, i.e., some entities would and should remain unmatched.

Consequently, the notion of stability here should be adjusted a little bit to capture the autonomy of both jobs to unilaterally fire employees and employees to just walk away. A matching is stable if

- there is no matched entity who prefers being unmatched over being with their current partner;
- there is no matched/filled job and unmatched candidate that would both prefer to be matched with each other over their current status;
- similarly, there is no unmatched job and matched candidate that would both prefer to be matched with each other over their current status;

- there is no matched job and matched candidate that would both prefer to be matched with each other over their current partners; and
- there is no unmatched job and unmatched candidate that would both prefer to be with each other over being unmatched.

(a) Prove that a stable pairing still exists in the case where we allow unmatched entities.

(HINT: You can approach this by introducing imaginary/virtual entities that jobs/candidates “match” if they are unmatched. How should you adjust the preference lists of jobs/candidates, including those of the newly introduced imaginary ones for this to work?)

(b) As you saw in the lecture, we may have different stable matchings. But interestingly, if an entity remains unmatched in one stable matching, it/she must remain unmatched in any other stable matching as well. Prove this fact by contradiction.

5 The Ranking List

Let’s study the stable matching problem a little bit quantitatively. Here we define the following notation: on day j , let $P_j(M)$ be the rank of the job that applicant M proposes to (where the first application on her list has rank 1 and the last has rank n). Also, let $R_j(W)$ be the total number of applicants that job W has rejected up through day $j - 1$ (i.e. not including the proposals on day j). Answer the following questions using the notation above.

- (a) Prove or disprove the following claim: $\sum_M P_j(M) - \sum_W R_j(W)$ is independent of j . If it is true, also give the value of $\sum_M P_j(M) - \sum_W R_j(W)$. The notation, \sum_M and \sum_W , simply means that we are summing over all applicants and all jobs.
- (b) Prove or disprove the following claim: one of the **applicants or jobs** must be matched to something that is ranked in the top half of their preference list. You may assume that n is even.