

CS70: Homework 2

Abhijay Bhatnagar

September 7, 2018

Sundry	1
Problem 1: Hit or Miss?	1
Problem 2: A Coin Game	2
Problem 3: Grid Induction	3
Problem 4: Stable Marriage	4
Problem 5: Optimal Partners	5
Problem 6: Examples or It's Impossible	6

Sundry

I primarily worked alone, but I talked through some of my proofs and consulted a friend in the course to see if they agree'd with my logic. Received assistance from:

George Matheos: georgematheos@berkeley.edu (310) 435-1961.

Problem 1: Hit or Miss?

- (a) Proof is correct. However, I might be a little more explicit about how the inductive hypothesis helps me make the final $n^2 + 2n + 1 \geq n + 1$ claim.
- (b) Proof is correct.
- (c) There is an error in the inductive step. The claim that you can represent $n + 1$ as $a + b$ where $0 < a, b \leq$ is not valid for all nonnegative n . Consider the base $n = 0$. It is impossible for two nonzero positive integers to sum to 1.

Problem 2: A Coin Game

Your "friend" Stanley Ford suggests you play the following game with him. You each start with a single stack of n coins. On each of your turns, you select one of your stacks of coins (that has at least two coins) and split it into two stacks, each with at least one coin. Your score for that turn is the product of the sizes of the two resulting stacks (for example, if you split a stack of 5 coins into a stack of 3 coins and a stack of 2 coins, your score would be $3 \cdot 2 = 6$). You continue taking turns until all your stacks have only one coin in them. Stan then plays the same game with his stack of n coins, and whoever ends up with the largest total score over all their turns wins.

Prove that no matter how you choose to split the stacks, your total score will always be $S_n = \frac{n(n-1)}{2}$. (This means that you and Stan will end up with the same score no matter what happens, so the game is rather pointless.)

Proof. We will proceed by Proof by Induction:

Base case Considering the base case of 1 coin, you can not split it, so the score always results in a score of 0. The formula holds for $n = 1$: $S_1 = \frac{2(2-1)}{2} = 0$.

Inductive Hypothesis Assume the claim holds true for all $1 < k \leq n$, i.e. the final score is always $S_k = \frac{k(k-1)}{2}$.

Inductive Step We prove the case for $n + 1$. With $n + 1$ original coins, we can split the stack into two smaller stacks of size a and b , where $a + b = n + 1$ and $1 \leq a, b \leq n$, yielding a score for the step of $a \cdot b$. Using the strong inductive hypothesis, we know the total scores of piles a, b will be S_a, S_b . This leaves the total score of $n + 1$ to be the following:

$$S_{n+1} = a \cdot b + \frac{a(a-1)}{2} + \frac{b(b-1)}{2} \quad (1)$$

$$= \frac{2ab + a(a-1) + b(b-1)}{2} \quad (2)$$

$$= \frac{a(a-1) + b(b-1) + 2ab}{2} \quad (3)$$

$$= \frac{(a+b)(a+b-1)}{2} \quad (4)$$

Substituting $a + b = n + 1$,

$$S_{n+1} = \frac{(n+1)(n)}{2}$$

Which was the desired result. By the principle of mathematical induction, the claim holds. \square

Problem 3: Grid Induction

There is a fairly obvious pattern that emerges when you consider that moving exactly one unit left or down per second from i, j will always take $i + j$ seconds to terminate.

Proof. Using a stronger hypothesis that he arrives at exactly $n = i + j$ seconds, we proceed by Proof by Induction on n :

Base case The base case of $(0,0)$ is already at the origin, therefore it took exactly 0 seconds. The base case holds.

Inductive Hypothesis Assume that for $n = k, k \in \mathbb{N}$ Pacman takes exactly k seconds to reach the origin.

Inductive Step For the case $n = k + 1$ The Pacman will be at some (i, j) such that $i + j = k + 1$. After one second, Pacman will move to either $(i, j - 1)$ or $(i - 1, j)$. WLOG, we can proceed with i as the decremented direction. From the inductive hypothesis, we know that for the position $(i - 1, j)$ where $(i - 1) + j = k$, Pacman will take exactly another k seconds to return to origin. Therefore, for some $i + j = k + 1$, Pacman will take a total of $k + 1$ to return to the origin. The sum $i + j$ is finite, thus, by the principle of induction the original theorem holds.

□

Problem 4: Stable Marriage

Consider a set of four men and four women with the following preferences:

men	preferences	women	preferences
A	1>2>3>4	1	D>A>B>C
B	1>3>2>4	2	A>B>C>D
C	1>3>2>4	3	A>B>C>D
D	3>1>2>4	4	A>B>D>C

(a) Day by day:

Days	Women	Proposals
1	1 2 3 4	A , B, C D
2	1 2 3 4	A D, B , C
3	1 2 3 4	A, D C B
4	1 2 3 4	D C, A B
5	1 2 3 4	D A B C

Final pairings: {1, D}, {2, A}, {3, B}, {4, C}

(b) Suppose we relax the time constraints so each man can propose whenever he wants. Prove the algorithm's output doesn't change.

Proof. Proceed by contradiction. Suppose the algorithm's output does change. Therefore, there must exist some stable pair (M^*, W) that differs from the original stable pair (M, W) . In the new algorithm, there will be a first time at $t=j$ a woman has rejected a man she previously accepted, i.e. W rejects M . This implies that at some time $t < j$, W has to have accepted some M^* . However, we know from the original algorithm that M^* will have proposed to some W^* before W . However, in order for M^* to later propose to W , that must imply W^* rejected M^* , which would imply that $t = j$ is not the first a woman made a new decision. This is a contradiction.

□

Problem 5: Optimal Partners

In the notes, we proved that the Stable Marriage Algorithm always outputs the male-optimal pairing. However, we never explicitly showed why it is guaranteed that putting every man with his best choice results in a pairing at all. Prove by contradiction that no two men can have the same optimal partner. (Note: your proof should not rely on the fact that the Stable Marriage Algorithm outputs the male-optimal pairing.)

Proof. We will proceed by contradiction. Suppose there are two men, M, M^* with the same optimal woman W . By definition, this means that both (M, W) and (M^*, W) are stable pairings, which implies that there must exist at least two solutions sets S_1, S_2 for each case.

Case 1: S_1 must contain two pairings (M, W) and (M^*, W') . Since we know W is M^* 's optimal woman, M^* must have proposed to W before W' and received a rejection. This implies that W prefers M over M^* .

Case 2: S_2 must contain two pairings (M^*, W) and (M, W') . Since we know W is M 's optimal woman, M must have proposed to W before W' and received a rejection. This implies that W prefers M^* over M .

The woman W cannot simultaneously prefer M^* over M and M over M^* , therefore there is a contradiction.

□

Problem 6: Examples or It's Impossible

Determine if each of the situations below is possible with the traditional propose-and-reject algorithm. If so, give an example with at least 3 men and 3 women. Otherwise, give a brief proof as to why it's impossible.

- (a) Every man gets his first choice. Example:

men	preferences	women	preferences
A	1>2>3	1	A>B>C
B	2>3>1	2	B>C>A
C	3>1>2	3	C>A>B

Day 1, each man proposes to his ideal partner, and they all accept.

- (b) Every woman gets her first choice, even though her first choice does not prefer her the most. Example:

men	preferences	women	preferences
A	3>1>2	1	A>B>C
B	1>2>3	2	B>C>A
C	1>3>2	3	C>A>B

- (c) Every woman gets her last choice. Example:

men	preferences	women	preferences
A	1>2>3	1	C>B>A
B	2>3>1	2	A>C>B
C	3>1>2	3	B>A>C

- (d) Every man gets his last choice. Impossible.

Proof. We proceed by contradiction. Suppose every man gets his last choice woman, this implies that the solution set has pairings $(M_1, W_1), \dots, (M_n, W_n)$. Since this is the last choice woman, there exists an (M_k, W_k) for which (M_1, W_k) and (M_k, W_1) are both stable pairs, both of which are more male optimal. The Stable Marriage algorithm is a male optimal algorithm, therefore the solution set that contains (M_1, W_k) and (M_k, W_1) would be created before the (M_1, W_1) and (M_k, W_k) , which creates a contradiction. \square

- (e) A man who is second on every woman's list gets his last choice. Example:

men	preferences	women	preferences
A	1>2>3	1	A>B>C
B	1>2>3	2	C>B>A
C	1>2>3	3	C>B>A