Problem Set 4 Solutions

Problem 1. **[10 points]**

Consider a robot which wanders on a plane. Our robot can move unit distance along any diagonal line (that is at an angle of ± 45 degrees or ± 135 degrees). If our robot starts at point $(0,0)$, can it ever reach the point $(1,1)$? Prove your answer.

Solution:

It cannot reach $(1,1)$. Let's model the robot's movements as a state machine and find an invariant which is preserved by every move of the robot. The states are pairs of coordinates (x,y), the initial state is $(0,0)$ and there are four possible movements:

$$(x,y) \;\leftarrow\; (x \pm \frac{\sqrt{2}}{2}, y \pm \frac{\sqrt{2}}{2})$$

An invariant of every reachable state (x,y) is that both x and y are integer multiples of $\sqrt{2}/2$. It holds for the initial state, and it is preserved by every move of our robot. Therefore, the machine can never reach a state $(1,1)$, as this state does not satisfy the invariant. If it did, i.e., if there was a natural number m s.t. $1 = m\sqrt{2}/2$ then we would have that $\sqrt{2} = 2/m$, and $\sqrt{2}$ would be a rational number. Contradiction.

Problem 2. **[20 points]**

We are given two buckets, A and B, a water hose, a receptacle, and a drain. The buckets and receptacle are initially empty. The buckets are labeled with their respectively capacities, positive integers a and b. The receptacle can be used to store an unlimited amount of water, but has no measurement markings. Excess water can be dumped into the drain. Among the possible moves are:

1. fill a bucket from the hose,
2. pour from the receptacle to a bucket until the bucket is full or the receptacle is empty, whichever happens first,
3. empty a bucket to the drain,
4. empty a bucket to the receptacle,
5. pour from A to B until either A is empty or B is full, whichever happens first,
6. pour from B to A until either B is empty or A is full, whichever happens first.

(a) [**4 points**]

Model this scenario with a state machine. (What are the states? How does a state change in response to a move?)

Solution:

The states are triples (x, y, z) which give the current amount of water in the bucket A, bucket B, and the receptacle, respectively. The initial state is $(0, 0, 0)$.

The moves make the following transitions:

1. fill a bucket from the hose

$$(x, y, z) \to (a, y, z) \text{ or } (x, b, z)$$

2. pour from the receptacle to a bucket until the bucket is full or the receptacle is empty, whichever happens first

$$(x, y, z) \to \begin{cases} (a, y, z - (a - x)) \text{ if pouring to A and } z \geq (a - x) \\ (x + z, y, 0) \text{ if pouring to A and } z < (a - x) \\ (x, b, z - (b - y)) \text{ if pouring to B and } z \geq (b - y) \\ (x, y + z, 0) \text{ if pouring to B and } z < (b - y) \end{cases}$$

3. empty a bucket to the drain

$$(x, y, z) \to (0, y, z) \text{ or } (x, 0, z)$$

4. empty a bucket to the receptacle

$$(x, y, z) \to (0, y, z + x) \text{ or } (x, 0, z + y)$$

5. pour from A to B until either A is empty or B is full, whichever happens first

$$(x, y, z) \to \begin{cases} (0, y + x, z) \text{ if } x < (b - y) \\ (x - (b - y), b, z) \text{ if } x \geq (b - y) \end{cases}$$

6. pour from B to A until either B is empty or A is full, whichever happens first

$$(x, y, z) \to \begin{cases} (0, y + x, z) \text{ if } x < (b - y) \\ (x - (b - y), b, z) \text{ if } x \geq (b - y) \end{cases}$$

(b) [**8 points**]

Prove that we can put $k \in \mathbb{N}$ gallons of water into the receptacle using the above operations iff $\gcd(a, b) \mid k$. *Hint*: Use the fact that if a, b are positive integers then there exist integers s, t such that $\gcd(a, b) = sa + tb$.

Solution:

We need to prove two facts (the "iff" statement has two directions):

1. If $\gcd(a, b) \mid k$ then we can put k gallons of water into the receptacle.
   Proof.
   Since there exist integers s, t s.t. $\gcd(a, b) = sa + tb$, then if $\gcd(a, b) \mid k$ then there exists an integer n s.t. $n * \gcd(a, b) = k$, and hence $n(sa + tb) = k$. Assume without loss of generality that $sa \geq tb$ (otherwise exchange the buckets in the following argument). Then we can fill the receptacle with k gallons if we first repeat ns times moves 1 and 4, filling the A bucket and pouring its content into the receptacle (note that if $sa \geq tb$ then $s \geq 0$). With this series of moves we will get from state $(0, 0, 0)$ to $(0, 0, nsa)$. Then, if $t = 0$, we are already done since $k = nsa$. If $t > 0$, we repeat nt times moves 1 and 4 but now using the bucket B. This gets us from $(0, 0, nsa)$ to $(0, 0, nsa + ntb) = (0, 0, k)$. If $t < 0$ we repeat $n * (-t)$ times moves 2 and 3 using the bucket B. At the end of this series we will be in state $(0, 0, nsa - n * (-t)b) = (0, 0, nsa + ntb) = (0, 0, k)$ ∎

2. If we can put k gallons of water into the receptacle then $\gcd(a, b) \mid k$.
   Proof.
   We show that it is an invariant of our state machine that $\gcd(a, b)$ divides x, y and z. It will prove in particular, that z is always a multiple of $\gcd(a, b)$
   Let's denote $\gcd(a, b)$ by c. The invariant is preserved by the initial state $(0, 0, 0)$.[1]. Each move preserves this invariant, because in each move the new values of x, y or z are always integer linear combinations of the previous values x, y, z or a, b, i.e., they are expressed by a formula $n_1x + n_2y + n_3z + n_4a + n_5b$ for some $(n_1, n_2, n_3, n_4, n_5) \in \mathbb{Z}^5$. For example in move 2, case 1, the new value of z is expressed by the above formula with $(n_1, n_2, n_3, n_4, n_5) = (1, 0, 1, -1, 0)$. Since c divides a and b, if c divides x, y, z before the move, it divides every such linear formula, and hence it divides the values of x, y and z after each move. ∎

(c) [8 points]

Let's restrict the valid moves to numbers 1 and 4 only. Prove that there exists a number n, such that for all integer $k \geq n$, we can fill the receptacle with k gallons of water iff $\gcd(a, b) \mid k$.

Solution:

Again, pick s, t s.t. $c = \gcd(a, b) = sa + bt$. If both $s, t \geq 0$ then $(s, t) = (0, 1)$ or $(1, 0)$, i.e., $a = b$ (check that in all other cases c would be greater than either a or b, which is impossible). If $a = b$ then $c = a = b$, and for each $k \geq c$, $c \mid k$, we can fill the receptacle by repeating moves 1 and 4 $k/c \in \mathbb{N}$ times (with any bucket, since $a = b$).

Clearly it cannot be that both $s, t \leq 0$ (or c would be a non-positive integer). So we are left with two cases: $t < 0 < s$ or $s < 0 < t$. Assume without loss of generality that $t < 0 < s$.

We show that if $k \geq n$ for $n = -tb^2/c$, and k is a multiple of c, then we can fill the receptacle with k gallons. Let's denote k as a multiple of b plus a "remainder" in the

[1] Remember that $(x \mid y) \iff (\exists_{b \in \mathbb{Z}}\ y = xb)$

following way:

$$k = k_1 b + k_2 c, \text{ where } k_1, k_2 \in \mathbb{N} \ \wedge \ 0 \le k_2 < \frac{b}{c}$$

We can represent k in this way since it is divisible by c. Notice also, that since $k \ge -tb^2/c$, then $k_1 \ge -tb/c$. Since $c = sa + bt$, we can rewrite k as follows:

$$\begin{aligned} k &= k_1 b + k_2 c \\ &= k_1 b + k_2(sa + bt) \\ &= (k_2 s)a + (k_1 + k_2 t)b \end{aligned}$$

Hence, as long as $k_1 \ge -tb/c$, both $(k_2 s)$ and $(k_1 + k_2 t)$ are non-negative integers. And hence, we can fill the receptacle by $(k_2 s)$ times pouring bucket A into it, and then $(k_1 + k_2 t)$ times bucket B, both just with moves 1 and 4.

Problem 3. [**25 points**]

Consider the following game for two players. The players alternate moves. A move consists of a pair (x, y) of natural numbers, subject to the constraint that no move may be $\ge$ to any previous move in both coordinates. The player who makes the $(0, 0)$ move is the loser.

At any point in the game, a valid move (a, b) is said to be *bounded* iff both a is bounded above by the x-coordinate of some previous move and b is bounded above by the y-coordinate of some previous move. For example, if the previous moves have been $(1, 3), (3, 1), (2, 2)$, then there are ten possible bounded moves: $\{(1, 2), (2, 1), (1, 1), (0, 3), (0, 2), (0, 1), (0, 0), (1, 0),$
$(2, 0), (3, 0)\}$. There is also an infinite number of possible unbounded moves at this point, namely, any move with one coordinate equal to 0 and the other larger than 3.

For any state S of the game after the first move, let $X(S)$ be the minimum of the x-coordinates of the previous moves, $Y(S)$ be the minimum of the y-coordinates, and $B(S)$ be the number of possible bounded moves. Define the *measure*, $M(S)$, of the state to be the triple $(X(S), Y(S), B(S))$.

Define the following ordering on triples of natural numbers:

$$(x, y, k) < (x', y', k') \iff [(x < x') \vee (x = x' \wedge y < y') \vee (x = x' \wedge y = y' \wedge k < k')$$

(a) [**10 points**] Prove that any nonempty subset of $\mathbb{N}^3$ has a least element under the above ordering.

Solution:

Let S be some nonempty subset of $\mathbb{N}^3$. Let $S_x = \{x \mid \exists_{y,z} \ (x, y, z) \in S\}$. Then $S_x \subset \mathbb{N}$, and so by the well-ordering principle, there is a smallest element x_{min} in S_x. Now take a set $S_y = \{y \mid \exists_z \ (x_{min}, y, z) \in S\}$. Since $S_y \subseteq \mathbb{N}$, and it is not empty, it has a smallest element y_{min}. Similarly, the set $S_z = \{z \mid (x_{min}, y_{min}, z) \in S\}$ has a smallest element z_{min}.

Clearly, $(x_{min}, y_{min}, z_{min})$ is the smallest element in S. If there was some smaller element (x', y', z') in S, then:

- If $x' < x_{min}$ then $x' \neq x_{min}$ would be the smallest element in S_x. Impossible.
- If $x' = x_{min}$ and $y' < y_{min}$ then $y' \neq y_{min}$ would be the smallest element in S_y. Impossible.
- If $x' = x_{min}$, $y' = y_{min}$ and $z' < z_{min}$ then $z' \neq z_{min}$ would be the smallest element in S_z. Impossible.

(b) [**10 points**] Conclude that the game always terminates, regardless of the players' strategy.

Solution:

We will show below that the measure $M(S) = (X(S), Y(S), B(S))$ is always decreasing according to the ordering in $\mathbb{N}^3$ defined above. On the other hand $M(S)$ cannot pass below $(0, 0, 0)$ because the minimum x coordinate, the minimum y coordinate and the number of bounded points cannot be negative. Therefore the game has to terminate.

Let's prove that measure $M(S)$ decreases with every move. Assume by contradiction that for some move leading from state $S = (x, y)$ to state $S' = (x', y')$, we have that $M(S') \geq M(S)$. There are three cases in which $M(S) \leq M(S')$ could be true:

1. $X(S) < X(S')$, which would mean that the minimum of the x-coordinates has increased, which is impossible (the minimum of a set to which we are adding some new element cannot increase).
2. $X(S) = X(S') \wedge Y(S) < Y(S')$, which would mean that the minimum of the y-coordinates has increased, which is impossible for the same reason as above.
3. $X(S) = X(S') \wedge Y(S) = Y(S') \wedge B(S) \leq B(S')$. Let $\{(x_1, y_1), \dots, (x_n, y_n)\}$ be the points that were picked up to move $S \to S'$ (i.e., the points that constitute state S), and let (x', y') be the new point. If $X(S) = X(S')$ and $Y(S) = Y(S')$, then $x' \geq \min_i\{x_i\} \wedge y' \geq \min_i\{y_i\}$.
   We first show that (x', y') must be a bounded point in state S. Assume otherwise. That would mean that either x' or y' were not bounded in state S. Without loss of generality, assume x' was not bounded, i.e., that $x' > x_i$ for all i. On the other hand, we know that $y' \geq y_j = \min_i\{y_i\}$. But then $y_j \leq (y')$ and $x_j < x'$ which would make (x', y') an illegal move. Hence (x', y') must be a bounded point of state S.
   Now we show that since (x', y') is bounded in S, state $S' = S \cup \{(x', y')\}$ must have fewer bounded points than S, i.e., that $B(S') < B(S)$. This will finish the proof that $X(S) = X(S') \wedge Y(S) = Y(S') \wedge B(S) \leq B(S')$ cannot be true.
   Notice that a point (x, y) is bounded in state $S = \{x_i, y_i\}$ iff it is a valid move and $x \leq \max_S\{x_i\}$ and $y \leq \max_S\{y_i\}$. Since, if (x', y') is bounded in S then the maximal x and y coordinates do not change from S to S'. Consequently, if $x \leq \max_{S'}\{x_i\} \wedge y \leq \max_{S'}\{y_i\}$ then $x \leq \max_S\{x_i\} \wedge y \leq \max_S\{y_i\}$. Also, if a point (x, y) is valid in S', it is also valid in S (it clearly follows from the fact that $S \subset S'$). From these two facts it follows that if a point is bounded in S', it is also bounded in S. In other words, the set $\mathcal{B}'$ of bounded points of S' is a

subset of $\mathcal{B}$, the set of bounded points of S. Since the latter is a finite set[2] and $\mathcal{B}' \subset \{\mathcal{B} \setminus (x', y')\}$, it follows that $B(S') = |\mathcal{B}'| < |\mathcal{B}| = B(S)$.

This proves that measure $M(S)$ always decreases and hence finishes the proof that the game always terminates.

(c) [**5 points**]

Does any player have a winning strategy? Prove your answer.

Solution:

The first player has two winning strategies:

Strategy 1: If the first player starts with $(1, 1)$ then whatever the second player picks, either $(0, n)$ or $(0, n)$ for some n, the first player will respond with, respectively, $(n, 0)$ or $(0, n)$. By symmetry, the first player will always have a move whenever the second player had a move, except when the second player picks $(0, 0)$ and loses.

Strategy 2: The second winning strategy is slightly more tricky: The first player starts by picking $(0, 2)$,[3] and then:

- If the 2nd player picks $(0, n)$ for any $n \geq 1$, 1st player responds with $(1, n - 1)$.
- If the 2nd player picks $(1, n)$ for any $n \geq 0$, 1st player responds with $(0, n + 1)$.

And the above procedure should be repeated for all consecutive moves. Notice that if player 2 had a valid move, so does player 1, unless player 2 picked $(0, 0)$ and lost.

Furthermore, there are no other winning strategies for player 1, because if she starts with any other legal move, player 2 will be able to adopt either of the two winning strategies and win for sure:

- If player 1 picks some (m, n) where $m, n \geq 1$ and $(m, n) \neq (1, 1)$, then player 2 can pick $(1, 1)$ next, and then use the first strategy to win against player 1.
- If player 1 picks some $(0, n)$ or $(n, 0)$ where $n \geq 3$, then player 2 can respectively pick $(0, 2)$ or $(2, 0)$ next, and then use the second strategy to win against player 1.
- If player 1 picks $(1, 0)$ or $(0, 1)$ then player 2 picks the other one and wins.

Problem 4. [**45 points**]

The Stable Marriage Problem was described in lecture. In this problem we generalize slightly to handle "bachelors." The setup is that there is a finite number, not necessarily equal, of men and women; each man has a strict (no ties) ranking of *some* of the women according to his preference. The women on a man's list are said to be *acceptable* to him. Likewise each woman has a ranking of the men who are acceptable to her.

[2] After the first move, the set of bounded points is always finite, because for whatever values x_{max}, y_{max}, there are finite number of pairs $(x, y) \in \mathbb{N}^2$ s.t. $x \leq x_{max}$ and $y \leq y_{max}$.

[3] Of course, picking $(2, 0)$ would be equivalent, and the following argument will hold for it as well, with x and y coordinates replaced

A *marriage set* is a bijection (matching) between a subset of the men and a subset of the women such that each man is matched to a woman he finds acceptable, and each woman is matched to a man she finds acceptable. In other words, a marriage set matches up men and women as "husband" and "wife" so that each man has exactly one wife, and each woman exactly one husband. Moreover, each wife is on her husband's list of ranked women, and each husband is on his wife's list of ranked men. Men and women not in the marriage set are called *bachelors*.

A husband in a marriage set is said to be *tempted* by a woman if (1) he prefers the woman to his wife, and (2) either the woman prefers him to her own husband, or the woman is a bachelor who finds him acceptable. Similarly, a wife may be tempted by a man. A marriage set is *stable* iff no husband is tempted by any woman and no wife is tempted by any man.

(a) [**5 points**] For all $n > 1$, describe a set of n men and m women for whom there is exactly one stable marriage set in which they are all married.

Solution:

One solution is to arrange preferences so that each woman is the first-choice of exactly one man, and that man is also her first choice. It's easy to see that when a man and a woman have each other as first choice, they will be a rogue couple unless they are married.

Another solution is to have all the men rank the women in the same order, and all the woman rank the men in the same order. In other words, all the lists held by the women are the same, and so are the lists held by the men. It is as if there was an "objective" ordering of both men that all women share and vice versa. Then the first-choice man must marry the first-choice woman, after which the second best man must marry the second woman, and so on. If they do not marry in this order then we can find the rogue couple as follows: Let's $x_1, \dots, x_n$ be the above ordering of men and $y_1, \dots, y_n$ be the ordering of women. Take the first man x_i who marries out of order ($i = \min\{k \mid x_k \text{ not married to } y_k\}$). Then x_i must have married "below his rank", i.e., he is married to y_j where $j > i$ (because each woman y_k for $k < i$ must be married to man x_k, or man x_i would not be the first who married out of order). The woman y_i that was "rightly his", must have married some man x_m who is also "below her status" (otherwise, x_m would be a man of a higher rank than x_i, who also married out of order). Notice that x_i and y_i form a rogue couple.

We can approach the Stable Marriage Problem with Bachelors using essentially the same protocol as in lecture (the only difference is the italicized phrase "*from a man she finds acceptable*" below):

The Mating Algorithm

The mating ritual takes place over several days. The following events happen each day:

Morning: Each man who has not yet crossed all the women off his rank list, "proposes" to the woman he most prefers among those remaining on his list.

Afternoon: Each woman who receives at least one proposal *from a man she finds acceptable*, says to her favorite suitor, "Maybe.... Come back tomorrow." To the other suitors, she says, "Cross me off your list." We say that on this afternoon, the woman is *engaged* to the man to whom she said "Maybe ... " (Some women may not be engaged to anybody on a given afternoon.)

Evening: Each man who hears a "Cross me off ... " message, crosses the woman who gave him the message off his list.

Termination condition: The algorithm terminates on the first afternoon when every man who proposed to a woman that morning ends up engaged to her in the afternoon. Then all the engaged couples get married.

(b) [**5 points**] Model the problem as a state machine. State clearly what the states and state transitions are. What is the start state?

Solution:

Just like in the original mating problem from the lecture, our state machine is $\mathcal{M} = (Q, q_0, \delta)$. States Q are all possible ways of crossing off girls' names on the boys' lists. The starting state q_0 is the state with no names crossed off on any list. The transitions δ lead from a state with a given set of boys' lists to a set in which some boys (possible none or all) have the same lists, and some have lists with the first girl crossed off.

In fact the boys whose lists remain the same are of three kinds:

- Those whose current list of girls is empty.
- Those whose current first girl does not appear as currently first on any other boy's list.
- Those whose current first girl appears on other boys lists as first, but these other boys are either unacceptable to this girl (not on her list of boys at all), or they have lower ranks on her list.

While the boys whose new lists have the first girl crossed off are of two kinds:

- Those who are not ranked on the list of boys of their current first girl (they are unacceptable to their current first girls).
- Those whose current first girl appears on other boys lists as first, and one of those other boys ranks higher on this girl's list.

(c) [**8 points**] Use derived variables to prove termination. How long does the protocol take to terminate?

Solution:

The same argument as in the original mating algorithm works here:

Let $f(q)$ be the number of uncrossed girl's names in state q. Initially, every boy's list has at most m uncrossed girl's names and so $f(q_0) \leq mn$. To prove that $f(q)$ is a decreasing derived variable, we must show that for every execution $q_0, q_1, \ldots$, we have $f(q_0) > f(q_1) > \ldots$. For this purpose, it is sufficient to show that $f(q) > f(q')$ for every transition $(q, q') \in \delta$.

In every transition (q, q'), since q does not satisfy the termination condition, we know that there is a man who proposes to some girl on that morning and does not end up engaged to her in the afternoon. This could only happen if she told him "Cross me off ... " on that afternoon, and hence he will cross her off in the evening. But this means that $f(q)$ decreases by at least 1.

Notice also, that $f(q) \geq 0$ for each q. Therefore, since there exists a decreasing derived variable $f(q)$ which take values in natural numbers, starts from $f(q_0) \leq mn$, and is bounded from below by 0, the algorithm terminates within nm steps.

(d) [**7 points**] Prove that the marriage set at the end of the protocol is stable.

Solution:

Again, the argument is the same as in the original mating algorithm: We use the same derived variables:

- Let $f_i(q)$ be the original rank that boy i assigned to the girl he is proposing to in state q. If the boy is not proposing to anyone, the value of the variable is $m + 1$.
- Let $g_i(q)$ be the original rank that girl i assigned to the boy to whom she says "Maybe ... " in state q. If nobody is proposing to her in state q, then the value of the variable is $n + 1$.

To prove the stability of the marriage set, we will need the following facts about the derived variables $f_i(q)$ and $g_i(q)$:

Facts

1. *f_i is a non-decreasing derived variable. As a boy crosses off names, the rank of the girl he proposes to increases; that is, he is forced to pursue girls that he ranks lower.*
2. *f_i is non-skipping. Each boy crosses off at most one name each day.*
3. *$f_i(q_0) = 1$. Initially, each boy proposes to his favorite gal.*
4. *The preceding two facts together with the Non-Skipping Theorem imply that if boy i marries eventually, then he must have at some point proposed to each girl that he ranked higher than his mate.*
5. *Similarly, if boy i never marries, he must have at some point proposed to each girl on his list.*
6. *g_i is a non-increasing derived variable. Each girl keeps stringing along her top choice by saying "Maybe" (holding g_i constant) until someone better comes along (decreasing g_i) or until the algorithm terminates.*

We now show that the algorithm produces a stable marriage set. Assume for the purpose of contradiction that there is a rogue couple $B - G$. Let G' be the girl that married B, and let B' be the boy who married G.

Since $B - G$ is a rogue couple, B must like G better than G'. Fact 4 implies that B proposed to G first. However, she must have said "Cross me off ... " to him. This, together with Fact 6, implies that G found someone she likes even better than

B. In particular, this implies that G likes B' better than B. But this contradicts the assumption that $B - G$ is a rogue couple.

(e) [**5 points**] Suppose that no person is unacceptable to more than one person of the opposite sex. Show that there can be a situation in which every stable marriage set leaves two bachelors of each sex. *Hint*: Find a situation involving exactly two men and two women.

Solution:
For example, take two men $M1, M2$ and two women $W1, W2$, and sadistically set their preferences as follows:

- M1 accepts only W1
- M2 accepts only W2
- W1 accepts only M2
- W2 accepts only M1

No mutually acceptable marriage is possible, hence the only stable marriage set is an empty set. Since it leaves two men and two women unmarried, it is true that every stable marriage leaves two bachelors of each sex.

Note that even though in every scenario an empty set is a stable marriage set which also leaves *all* men and women unmarried, it is not true for every scenario that *every* stable marriage set has this property.

(f) [**8 points**] Suppose that the number of women and men is the same, and that no person is unacceptable to more than one person of the opposite sex. Show that the protocol leads to a stable marriage set where at most two men are bachelors.

Solution:
We first prove a helpful fact:
If a man finds a woman acceptable and vice versa, then it cannot be that both end up as bachelors.

Proof. Suppose for contradiction that M and W are bachelors. If M finds W acceptable, he is either married to her or proposed to her at some point (facts 4 and 5 in part 4d). Since he is a bachelor now, he must have proposed to her at some point, and she must have rejected him. But since M is acceptable to her, she could have rejected him only because there was some man she was engaged to at that time. But once a woman is engaged to somebody, she has to marry (not necessarily the same person, but someone). Hence we have a contradiction. ■

Now assume that there are at least 3 bachelors of one sex, say men.

- Since the number of men is equal to the number of women, if there are 3 male bachelors, there are 3 female bachelors.
- Pick a bachelor woman W1 (call the other two W2 and W3). She must be unacceptable to at most one of the bachelors. Call the other two bachelors, those that find W acceptable, M2 and M3.

- If W1 found either M2 or M3 acceptable, then she would not not end up a bachelor (from the fact argued above, and because both M1 and M2 are bachelors). So she finds them unacceptable.
- Since no man can be unacceptable to more than one woman, woman W2 (same holds for W3) must find both M2 and M3 acceptable.
- Since no woman can be unacceptable to more than one man, W2 must be acceptable to either M2 or M3 (same for W3). Without loss of generality say that M2 finds W2 acceptable.
- By the fact argued above, it cannot be that both M2 and W2 end up as bachelors.

(g) [**7 points**]

Let's modify the mating algorithm above by introducing the following change to the procedure for the Afternoon events. On each afternoon, all the women say "Maybe … " as before, but only one woman says "Cross me off your list" and she says it to only one (but any one) among the suitors she would have said it to in the protocol above.

In particular, on each afternoon at most one "Cross me off your list" is spoken. Notice that now the state of the men is no longer uniquely determined by which afternoon it is. Briefly indicate whether this refined protocol always yields a stable final marriage set. When there is a final marriage set, is it necessarily the same as the one from the original protocol?

Solution:

Q: Does the new protocol always yield stable final marriage set?
A: Yes. The same argument as in part 4c applies here to show that the new one-at-a-time protocol terminates (it runs slower, but it terminates eventually). And also the same argument as in part 4d applies here to show the one-at-a-time protocol leads to a stable marriage set. Notice that all the facts 1-6 from part 4d are still true in the new protocol.

Q: Is the final marriage set of the new protocol the same as the one from the original protocol?
A: Yes. The two protocols produce exactly the same marriage sets, because both of them lead to a marriage set which is man-optimal. You can show that the morning-afternoon-evening protocol is man-optimal in the same way as it is shown in the lecture notes (theorem 2.9) for the original mating problem from the lecture. Moreover, the same argument applies to the one-at-at-time protocol presented here. Since there is exactly one man-optimal matching (every man marries the first woman on his list), the two protocols find the same matchings.