Max Shi

I pledge my honor that I have abided by the Stevens Honor System.
Max Shi

CS 135 Problem Set 7

1. Proof by contradiction:
No boy received exactly one proposal in the process.
- Case 1: Each boy received 2 or more proposals in the process. i.e. on the last day, no boy received their first proposal. If this is the case, then the last boy to get a proposal on the last day must have received a proposal. However, because no boy received their first proposal, this boy must have already had a proposal. Therefore, the boy must reject one of these proposals, and the algorithm must continue, which is a contradiction that this was the last day.
- Case 2: At least one boy received 0 proposals in the process. The algorithm can only terminate if all N girls are attached to N unique boys at the end of the process. Thus, if the algorithm terminated with a boy receiving 0 proposals, it means N girls were paired with N-1 boys, which is a contradiction.

Thus, it follows that at least one boy receives exactly one proposal.

2. Each girl has a list of N boys to propose to. From the last problem, on the last day, there is one previously unattached boy who received his first proposal. Therefore, on the turn before the last turn, each girl must have at most made $N-1$ proposals, because none of them proposed to the unattached boy. Because there are N girls, this means that on the turn before the last, there had been at most $N(N-1)$ proposals. In this scenario, there can only be one proposal on the last day, as more than one proposal would suggest there are $x > 1$ previously unattached boy, this makes the prior number of proposals at most $N(N-x)+x$, which is N^2-Nx+x. In our scenario, with one proposal on the last day, that quantity is $N(N-1)+1$, which is N^2-N+1. Because N is >1, and $2<x$, isolating the different terms gives, $(-Nx+x)$ and $(-N+1)$ respectively. The former can be factored as $(-N+1)x$, and $N>1$ means $-N+1$ is negative, therefore $(-Nx+x)<(-N+1)$, and $N(N-1)+1$ is the upper bound on proposals.

3. N proposals are made on the first day. To get the upper bound on number of days, we assume the minimum amount of proposals for each subsequent day, which is 1. Thus, after the first day, it takes $N(N-1)+1-N$ days to make the maximum amount of proposals, which is $N(N-1)+1$ from the previous problem. Thus, expanded out, this quantity is N^2-2N+1 for all subsequent days, Adding on the first day gives N^2-2N+2, thus giving us the upper bound on the number of days this algorithm takes to complete

4.

Day 1:	Day 2:	Day 3:	Day 4:	Day 5
$G_1 \longrightarrow B_1$	$G_1 \rightarrow B_1$	$G_1 \rightarrow B_1$	$G_1 \rightarrow B_1$	$G_1 \rightarrow B_1$
$G_2 \longrightarrow B_2$	$G_2 \rightarrow B_2$	$G_2 \rightarrow B_2$	$G_2 \rightarrow B_2$	$G_2 \rightarrow B_2$
$G_3 \longrightarrow B_3$	$G_3 \rightarrow B_3$	$G_3 \rightarrow B_3$	$G_3 \rightarrow B_3$	$G_3 \rightarrow B_3$
$G_4 \longrightarrow B_4$	$G_4 \rightarrow B_4$	$G_4 \rightarrow B_4$	$G_4 \rightarrow B_4$	$G_4 \rightarrow B_4$
G_5 B_5	G_5 B_5	G_5 B_5	G_5 B_5	$G_5 \rightarrow B_5$

5 days to terminate algorithm. This is consistent with problem 3, as $N=5$, and $N^2-2N+2 = 25-10+2 = 17$, and $5<17$.

5. The algorithm would run N^2-2N+2 times if preferences are as follows:
For every girl i, $1 \le i \le N$, their list is ranked as follows:

G_i: $B_i, B_{i+1} \dots B_{N-1}, B_1, B_2 \dots B_N$.

As in, the ith girl up to but not including the Nth girl has the list starting with the ith boy and ascending either until the Nth boy or reaching the N-1 boy and starting from 1 until all boys except the Nth are chosen, and the Nth boy is last.
The Nth girl should then have the same list as the 1st girl
The boys' preferences should go as such:
the ith boy except the last should have list: (starting at $i+1$, going down to G_1, looping around to top)

$G_{i+1}, G_i, G_{i-1}, \dots, G_1, G_N, G_{N-1} \dots G_{i+2}$.

G_1

This results in one rejection/day, ending with the first girl proposing to the last boy.