**Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**CS105 Homework #1**

1. Use the Gale-Shapley algorithm for stable matching to compute a stable matching using the following rankings. In each row, circle the match found for the corresponding person using the men-proposing version of the algorithm, and put a rectangle around the match found using the women-proposing version of the algorithm.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Chandler |  | Janice | Julie | Kathy | Monica | Phoebe | Rachel | Mona | Emily |
| Gunter |  | Phoebe | Emily | Julie | Rachel | Janice | Mona | Kathy | Monica |
| Joey |  | Rachel | Mona | Emily | Phoebe | Julie | Janice | Kathy | Monica |
| Joshua |  | Emily | Kathy | Mona | Julie | Phoebe | Janice | Rachel | Monica |
| Pete |  | Rachel | Phoebe | Mona | Janice | Emily | Julie | Kathy | Monica |
| Richard |  | Monica | Phoebe | Julie | Rachel | Mona | Emily | Kathy | Janice |
| Ross |  | Kathy | Julie | Rachel | Janice | Monica | Mona | Phoebe | Emily |
| Tag |  | Phoebe | Monica | Rachel | Kathy | Emily | Janice | Julie | Mona |
|  |  |  |  |  |  |  |  |  |  |
| Emily |  | Richard | Chandler | Joey | Ross | Gunter | Joshua | Tag | Pete |
| Janice |  | Joshua | Tag | Chandler | Gunter | Richard | Ross | Joey | Pete |
| Julie |  | Joey | Gunter | Joshua | Tag | Chandler | Ross | Pete | Richard |
| Kathy |  | Pete | Joey | Ross | Gunter | Tag | Richard | Joshua | Chandler |
| Mona |  | Tag | Gunter | Chandler | Pete | Joshua | Joey | Richard | Ross |
| Monica |  | Pete | Tag | Joshua | Richard | Ross | Chandler | Gunter | Joey |
| Phoebe |  | Chandler | Gunter | Tag | Richard | Ross | Joey | Pete | Joshua |
| Rachel |  | Tag | Pete | Joey | Joshua | Gunter | Richard | Chandler | Ross |

1. Reconsider the stopping problem that Ross faced when the fortune teller told him that he would meet 4 women in his future. Suppose we wanted Ross to end up with the worst possible match as frequently as possible.
	1. How might we change the best match algorithm we used previously to find the worst match? How often will Ross get his worst match?
	2. If instead, we used the same algorithm that we used for finding the best match, what value would we use for k to generate the worst match most often? How often will Ross get his worst match?
2. Suppose you know that your professor gives pop quizzes at random intervals, but overall, about 1 class in 5. If a student misses 4 classes in the course of the semester, what is the probability that the student will miss at least one quiz?
3. Reconsider the pirates’ problem that we did in class. Suppose that the pirates changed the rules so that for a proposal to pass, it required at least ¾ of the pirates to vote for it. What proposal should the first pirate make? Will it pass?