**Time Spent:** 2 hours

**Collaborators and Resources:**

---

## Problem 2 (b)

In class we have discussed rather inefficient algorithms where men propose to their highest preference and women (tentatively, for the present time in the algorithm's running) accept their best proposal so far and how this avoids instability pairs (m, w') and (m', w) wherein m prefers w and w prefers m. The crucial difference between the situation described and Problem 2 is essentially that we do not split the students into two equally sized bipartite groups. Instead what we have is a graph in which in-group pairings are possible and as a result there is always going to be a greater number of possible matchings.

Below is my example of a preference list ...

| | 1st | 2nd | 3rd |
|---|---|---|---|
| Adam | Y | Z | X |
| Xavier | Z | Y | A |
| Yolanda | X | Z | A |
| Zeus | Y | X | A |

To demonstrate that no stable matching exists, we can just iterate over all the possible matchings: In the (A, X), (Y, Z) case Yolanda prefers Xavier to Zeus and Xavier prefers Yolanda to Adam. In the (X, Y), (A, Z) case Xavier prefers Zeus and Zeus prefers Xavier. In the (X, Z), (A, Y) case Yolanda prefers Zeus and Zeus prefers Yolanda. As a result, each possible pair of pairings represents an instability, so any algorithm is not going to find a stable matching.