Instructions: To solve these problems, you are allowed to consult your classmates, as well as the class textbook (*Algorithmic Game Theory*) and notes, but no other sources. As with all future homeworks, you are allowed to write up your solutions in pairs, and turn in only one copy for each pair. Please write clearly and concisely.

[20 points] Consider the following group formation game. There are \( n \) players, and \( m \) possible groups (e.g., clubs, political parties, etc). The strategy of a player is just to choose which single group they will join, so the strategy set of player \( i \) is \( S_i = \{1, \ldots, m\} \). Thus, an outcome of this game corresponds to a partition of the players; let \( A_k \) be the set of players who choose group \( k \).

For every pair of players \( i \) and \( j \) there is a number \( w_{ij} \geq 0 \) signifying how much \( i \) and \( j \) want to be in the same group. You can think of this as a complete graph with edges of weight \( w_{ij} \); assume that \( w_{ij} = w_{ji} \).

The utility of player \( i \) is defined to be \( u_i(s) = \sum_{j | s_j = s_i} w_{ij} \).

(a) Prove that this is an exact potential game, and thus a pure Nash equilibrium always exists, and best-response dynamics converge.

(b) Prove that the Price of Stability for this game is always equal to 1.

(c) Give an example for which the Price of Anarchy is equal to \( m \).

The game above is actually kind of silly: why don’t all the players simply join the same group, and obtain the best possible utility? Consider a more general version of this game, in which the players are not indifferent between groups. Specifically, for every group \( k \) and player \( i \), there is a number \( x_{ik} \geq 0 \) saying how much player \( i \) wants to be in group \( k \). The utility of player \( i \) is now \( u_i(s) = x_{is_i} + \sum_{j | s_j = s_i} w_{ij} \).

(d) Is this still an exact potential game? Show that unlike for the previous case, the price of stability here can approach 2, and prove that it can never be higher than 2.