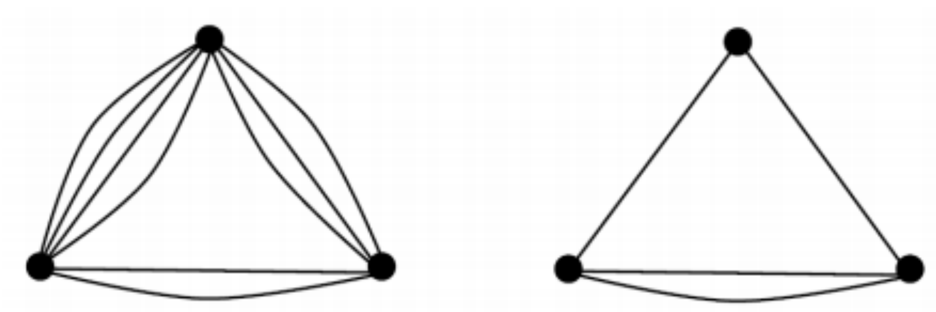
\usepackage*graphicx*

# Assignment 3

## Question 1

Graph NIM. The game is played on a 3-vertex graph, and between each pair of vertices is any number of edges; there are no loops. Suppose the players are named A and B, and it is decided that A makes the first move. Each move includes two steps: (1) choose a vertex, (2) choose any nonzero number of the edges incident to that vertex (and only to that vertex) to delete and delete them. The winner of the game is the player who removes the final edge(s).

Example: The gure on the left below shows a possible starting game. The one on the right is the result of a rst move. In this first move player A chose the vertex at the top, and removed ve edges incident to it, leaving one edge from each lower vertex to the top vertex.



Confirm or deny, with proof, whether player A (the first player) can always win if she employs a particular strategy for each move. Note that this strategy must be independent of B's strategy, and there may be a constraint on the initial state of the graph on which the game is played.

*Claim*: There exists a strategy that will ensure the victory of player A as long as player A always goes first.

*Proof*: Lets define a new board where vertex A is the top most point, vertex B is the left most point, and vertex C is the right most point. See the final page of this packet and refer to the question 1 figure to describe that will be played below.

Now lets create a new game board with a random number of verticies that is greater than 2 and none of the edge counts can be equal to another edge count to provide our constraint, also in order to guarentee victory, Player 1 must always go first. For this example lets say there are 55 edges between point A and B, 22 edges between point A and C, and 10 edges between points C and B. The strategy for player A to guarentee a win should go as follows: For each of player A's moves, player A needs to choose a vertex that will reduce the number of edges that will make all of the edges between each available verticie equal to the lowest number of edges between 2 verticies. For example in the game board I have created, each edge would be equal to 10 after player first A's move. Regardless of what player B does with it's turn player A will have another turn, lets create a random move by player B where point A was chosen to remove edges, lets say after player B's move the game board goes as follows, the edge count between vertex A and B is equal to 5, the edge count between point A and C is equal to 2, and the edge count between point B and C is equal to 10. For player A's next move, player A must pick the vertex that will make all edges between 2 verticies equal to the lowest number of edges between 2 verticies, therefore player A must choose point B and make all edges between any 2 verticie equal to 2. If we follow this pattern this proves that whatever player B chooses at this point will result in one at least one more vialable move before player B is forced to make a move that guarentees a move for player A to make that will claim ensure its victory.

# Question 2

Please prove that in every flock, if a chicken is pecked, then it is pecked by a queen.

*Claim*: If a chicken is pecked, it was pecked by a queen.

*Proof*: Lets say that a flock is represented by F = (C, $\rightarrow$) where C is a chicken and $\rightarrow$ is a peck. If you refer to the question 2 figure on the final page of this packet, we will define a simple flock that will show a generic chicken relationship. On the image provided I have defined the queen chicken in this flock and lets call this chicken $C_1$ and lets let all other chickens $C_i \in$ F, by observation we know that $C_1$ has some sort of relationship with all other $C_i$ , either $C_1 \rightarrow C_i$ or $C_i \rightarrow C_1$ . After observing the provided image image we notice that we can apply a theorem, In any flock there is a queen. We can show this theorem if we let q be equivelant to a queen and q $\in$ C such that for every x (x being the smaller circled subset) $\in$ C{q} either q $\rightarrow$ x or (x $\rightarrow$) there is a y = C{q, x} (y being the the subset of the whole flock) such that q $\rightarrow$ y and y $\rightarrow$ x. If we analyize this theorem we can conclude that in any flock if a chicken is pecked it was pecked by a queen because of the relationship in the whole flock of x $\rightarrow$ $C_1$ $\rightarrow$ y.

# Question 3

- Please prove that is impossible for all chickens to be a queens with exaclty 4 chickens.

*Claim*: It is impossible for all chickens to be a queens with exaclty 4 chickens.

*Proof*: If you observe the figure under question 3 figure on the final page of the packet you will see that for any variation of 4 chickens and their pecking orders. It is impossible for all chickens to be queens, and here's why. On my image I have depicted every possiblity of 4 chickens and the pecking possibilies via the score of a chicken. The score of a chicken is reflected on the amount of times it gets pecked by all other chickens, limiting one peck from a chicken. Using this score, we see that they all add to 6. This guarentees that we have seen the maximum number of pecks for a flock of 4 chickens.

A queen is defined by: For a single chicken lets say chicken A, chicken A is a queen if this chicken pecks all other chickens either directly or if chicken A pecks another chicken (lets say chicken B) If chicken B pecks another chicken (lets say chicken D) that chicken D must be reached within 2 pecks from chicken A lets call this a indirect peck. For chicken A to be a queen all other chickens must be pecked using either a direct peck, or an indirect peck. If all chickens are pecked, chicken A is a queen.

Applying the pecking order used above, I prove it is impossible for all 4 chickens to be queens by showing every possibily of pecking orders.