CS240: Homework 9

Sahit Mandala

December 2, 2015

Problem 1

Consider the relation R on V defined by mutual reachability. Suppose that uRv and vRw for some u,v,w in V. We want to show that R is an equivalence relation.

Symmetric: For any x in V, notice that xRx since any node x is reachable from itself, x, using the trivial path with no edges ; hence, x is mutually reachable from x, so xRx, proving R is symmetric.

Reflexivity: Consider uRv. That is, u is mutually reachable from v. So there exists a directed path from u to v and a directed path from v to u. From this, it trivially follows from the definition that v is mutually reachable from u, so vRu, proving R is reflexive.

Transitivity: Consider uRv and vRw. Then we know that u and v as well as v and w are each mutually reachable. So we know there exists a directed path from u to v, say $P1 = (u, x_1), ..., (x_n, v)$, and a directed path from v to w, say $P2 = (v, y_1), ..., (y_m, w)$. Now consider the path P_3 constructed by appending P_2 to P_1 (.e.g. $P_3 = P_1 \cup P_2 = ((u, x_1), ..., (x_n, v), (v, y_1), ..., (y_m, w))$. P_3 is a directed path from u to w, which shows that w is reachable from u. Similarly, we know there exists a directed path from v to u, say $P_1 = ((v, x_1), ..., (x_n, u))$, and a directed path from w to v, say $P2 = ((w, y_1), ..., (y_m, v))$. Now consider the path P_3 constructed by appending P_2 to P_1 (.e.g. $P_3 = P_1 \cup P_2 = ((w, x_1), ..., (x_n, v), (v, y_1), ..., (y_m, u))$. P_3 is a directed path from w to u, which shows that u is reachable from w. Hence, overall, we have show that u is mutually reachable to w, so uRw, showing reflexivity.

Overall, this shows that R is a equivalence relation on V.

Problem 2

Let some $G = (V, E_G)$,$H = (V, E_H)$ be given as defined in the problem. If G is connected, we are done, so now suppose that G is not connected. We want to show now that H is connected. So let some $u, v \in V$ be given with $u \neq v$. There are 2 cases we will consider:

Case 1: u,v were not connected in G

This implies that there is no edge (u, v) in G, or they would be connected in G. Hence, since $u \neq v$ and $(u, v) \notin E_G$, $(u, v) \in E_H$. Hence, u and v are clearly connected in H.

Case 2: u,v were connected in G

In this case, consider u. Because G is not connected, we can choose some vertex $x \neq u \in V$ s.t. u is not connected to x in G (otherwise, every node would be connected to every other node through u). So clearly, there is no edge from u to x, so $(u, x) \notin E_G$; hence, $(u, x) \in E_H$ by definition of H. Furthermore, we can see that v is not connected to x in G since, if x is connected to v, it follows that u would also be connected to x (since we can then construct an aggregated path from u to v and then v to x), contradicted our definition on x. So v is not connected to x in G, implying there is no edge from v to x in G, so $(v, x) \notin E_G$. Hence, $(v, x) \in E_H$ by definition. Because u is connected to x in H and v is connected to x in H, we see that u and v must be connected to each other (just consider the path $((u, x), (x, v))$)

Overall, this shows that u,v are connected in H. Because u,v are arbitary, it follows that every vertex is connected to every other one, implying that H is connected.

Hence, we have shown that atleast one of G, H is connected.

Problem 3

Part a

See attached

Part b

i)

K_n always has n vertices. As for edges, we make it a counting problem. The first node is connected to $n-1$ nodes, so we can count $n-1$ edges out of the first node. The second node also has $n-1$ vertices out of it, but we already counted the one to the first node. So there are $n-1-1 = n-2$ distinct edges from the second node. Again, the 3rd has $n-1-2 = n-3$ distinct edges, ignoring the edges to the 1st, 2nd node. If we continued this and summed up all the distinct nodes, we see that the total is $n-1+n-2+n-3+...+1+0 = \frac{n(n-1)}{2}$ edges. So there are $\frac{n(n-1)}{2}$ edges.

ii)

$K_{n,m}$ has 2 separate subsets with n and m elements respectively, so there are a total of $n+m$ verticies.

We know that every node in the first subset (n elements) is connected to every element in the other subset (with m elements). So each of the n elements in the first subset has m edges, one to each of the m elements in the other subset. So the total number of edges is $m * n$.

iii)

C_n has n vertices by construction. Furthermore, we know that by construction of the edges $\{v_1, v_2\}, \{v_2, v_3\}, ..., \{v_{n-1}, v_n\}, \{$ that we can assign each edge to a vertices by considering the "first" vertex on the edge. So we can pair off $v_1 \to \{v_1, v_2\}, v_2 \to \{v_2, v_3\}, ..., v_{n-1} \to \{v_{n-1}, v_n\}, v_n \to \{v_n, v_1\}$. By the definition's construction of edges, every vertex appears once as the "starting" vertex for exactly one edge, making this a bijective pairing. Point it, there are the same number of edges as vertices, so there are also n edges.

Problem 4

Part a

(FWGC,0) (FWG,C) (FWC,G) (FGC,W) (FG,WC)
(0, FWGC) (C, FWG) (G, FWC) (W, FGC) (WC, FG)

Part b

See attached

Part c

Every node in the graph corresponds to an allowable state in the puzzle by construction from part a. Furthermore, every edge (u,v) corresponds to an allowable moves in the puzzle, where the farmer can make a particular trip to switch from state u to state v (or vice versa). (FWGC,0) corresponds to the starting state (everyone on one side of the river, wanting to cross) while (0,FWGC) corresponds to the ideal end state (everyone has crossed the river, on other side). A path from (FWGC,0) to (0,FWGC) would correspond to a particular list of allowable moves/trips the farmer could make in the puzzle resulting in him getting everyone from one side to another.

Part d

To find 2 different solutions, we shall consider 2 unique paths from the starting state to ending state:

$(FWGC, \emptyset) \rightarrow (WC, FG) \rightarrow (FWC, G) \rightarrow (W, FGC) \rightarrow (FWG, C) \rightarrow (G, FWC) \rightarrow (FG, WC) \rightarrow (\emptyset, FWGC)$

(This corresponds to taking goat right, returning left, taking cabbage right, taking goat left, taking wolf right, returning left, and finally taking goat right)

$(FWGC, 0) \rightarrow (WC, FG) \rightarrow (FWC, G) \rightarrow (C, FWG) \rightarrow (FGC, W) \rightarrow (G, FWC) \rightarrow (FG, WC) \rightarrow (\emptyset, FWGC)$

(This corresponds to taking goat right, returning left, taking wolf right, taking goat left, taking cabbage right, returning left, and finally taking goat right)

Part e

On the first solution, we see that there are 4 animal crossings (goat right; goat left; wolf right;goat right) so that would be a total total of $4.

On the second solution, we see that there are 4 animal crossings (goat right; wolf right; goat left;goat right) so that would be a total total of $4.

Both methods require the same number of dollars to bring everyone to the other side, so either solution is feasible.