ECE608 chapter 26 problems

1) 26.1-4
2) 26.1-6
3) 26.2-11
4) 26.3-3
5) 26-4

function EDGE-CONNECTIVITY(G)
  select any vertex $u \in V$
  for each vertex $v \in V - \{u\}$ **do**
    set up he flow network G_{uv} as described
    find the maximum flow f_{uv} on G_{uv}
  end for
  return the minimum of the $|V| - 1$ max-flow values: $\min_{v \in V - \{u\}} |f_{uv}|$
end function

Figure 1: Edge connectivity 26.2-11

(26.1-4) To show that the flows form a convex set, we need to show that for any two flows f_1 and f_2 and any $0 \leq \alpha \leq 1$, we have that $\alpha f_1 + (1-\alpha) f_2$ is also a flow.
Firstly we have for any $u, v \in V$, $\alpha f_1(u,v) + (1-\alpha) f_2(u,v) \geq 0$ since f_1 and f_2 are flows.
Since $f_1(u,v) \leq c(u,v)$ and $f_2(u,v) \leq c(u,v)$, we have that

$$\alpha f_1(u,v) + (1-\alpha) f_2(u,v) \leq \alpha c(u,v) + (1-\alpha) c(u,v) = c(u,v)$$

For flow conservation–since f_1 and f_2 obey flow conservation–for any $\sum_{v \in V} f_1(u,v) = 0$ and $\sum_{v \in V} f_2(u,v) = 0$ for any $u \in V - \{s,t\}$. Thus,

$$\sum_{v \in V} (\alpha f_1(u,v) + (1-\alpha) f_2(u,v)) = \alpha \sum_{v \in V} f_1(u,v) + (1-\alpha) \sum_{v \in V} f_2(u,v) = \alpha \times 0 + (1-\alpha) \times 0 = 0$$

So, any convex combination of flows is also a valid flow. Hence, flows form a convex set.

(26.1-6) Create a vertex for each corner, and if there is a street between corners u and v, create directed edges (u,v) and (v,u). Set the capacity of each edge to 1. Let the source be corner on which the professor's house sits, and let the sink be the corner on which the school is located. We wish to find a flow of value 2 that also has the property that $f(u,v)$ is an integer for all vertices u and v. Such a flow represents two edge-disjoint paths from the house to the school.

(26.2-11) For any two vertices u and v in G, we can define a flow network G_{uv} consisting of the directed version of G with all edge capacities set to 1, $s = u$, and $t = v$. (G_{uv} has $O(V)$ vertices–actually, $|V|$–and $O(E)$ edges, as required. We want all capacities to be 1 so that the number of edges crossing a cut equals the capacity of the cut.) Let f_{uv} denote a maximum flow in G_{uv}. We claim that for any $u \in V$, the edge connectivity k equals $\min_{v \in V - u} |f_{uv}|$. We show that this claim holds. Assuming that it holds, we can find k as shown in Fig. 1.
The claim follows from the max-flow min-cut theorem and how we chose capacities so that the capacity of a cut is the number of edges crossing it. We prove that

$k = \min |f_{uv}|$, for any $u \in V$ by showing separately that k is at least this minimum and that k is at most this minimum.

- Proof that $k \geq \min_{v \in V-\{u\}} |f_{uv}|$: Let $m = \min_{v \in V-\{u\}} |f_{uv}|$. Suppose we remove only $m-1$ edges from G. For any vertex v, by the max-flow min-cut theorem, u and v are still connected. (The max flow from u to v is at least m, hence any cut separating u from v has capacity at least m, which means at least m edges cross any such cut. Thus at least one edge is left crossing the cut when we remove $m-1$ edges.) Thus every node is connected to u, which implies that the graph is still connected. So at least m edges must be removed to disconnect the graph–i.e., $k \geq \min_{v \in V-\{u\}} |f_{uv}|$.

- Proof that $k \leq \min_{v \in V-\{u\}} |f_{uv}|$: Consider a vertex v with the minimum $|f_{uv}|$. By the max-flow min-cut theorem, there is a cut of capacity $|f_{uv}|$ separating u and v. Since all edge capacities are 1, exactly $|f_{uv}|$ edges cross this cut. If these edges are removed, there is no path from u to v, and so our graph becomes disconnected. Hence $k \leq \min_{v \in V-\{u\}} |f_{uv}|$.

- Thus, the claim that k= $\min |f_{uv}|$, for any $u \in V$ is true.

(26.3-3) An augmenting path is a simple path $s \rightsquigarrow t$ in the residual graph G_f. Since G is bipartite, it has no edges between vertices in L and no edges between vertices in R. So the flow network G and G_f also do not have any such edges. The only edges involving s or t connect s to L and R to t. Thus any augmenting path must be of the form $s \to L \to R \to \cdots \to L \to R \to t$, crossing back and forth between L and R at most as many times as it can do so without using a vertex twice. It contains s, t, and equal numbers of distinct vertices from L and R–at most $2 + 2 \times \min(|L|, |R|)$ vertices in all. The length of an augmenting path (i.e., its number of edges) is thus bounded above by $2 \times \min(|L|, |R|) + 1$.

(26-4) a) Just execute one iteration of the Ford-Fulkerson algorithm. The edge (u, v) in E with increased capacity ensures that the edge (u, v) is in the residual graph. So look for an augmenting path and update the flow if a path is found.
Time: $O(V + E) = O(E)$ if we find the augmenting path with either depth-first or breadth-first search. To see that only one iteration is needed, consider separately the cases in which (u, v) is or is not an edge that crosses a minimum cut. If (u, v) does not cross a minimum cut, then increasing its capacity does not change the capacity of any minimum cut, and hence the value of the maximum flow does not change. If (u, v) does cross a minimum cut, then increasing its capacity by 1 increases the capacity of that minimum cut by 1, and hence possibly the value of the maximum flow by 1. In this case, there is either no augmenting path (in which case there was some other minimum cut that (u, v) does not cross), or the augmenting path increases flow by 1. No matter what, one iteration of Ford-Fulkerson suffices.
b) Let f be the maximum flow before reducing $c(u, v)$. If $f(u, v) = 0$, we don't need to do anything. If $f(u, v) > 0$, we will need to update the maximum flow. Assume from

now on that $f(u,v) > 0$, which in turn implies that $f(u,v) \geq 1$. Define $f'(x,y) = f(x,y)$ for all $x, y \in V$, except that $f'(u,v) = f(u,v) - 1$. Although f obeys all capacity contraints, even after $c(u,v)$ has been reduced, it is not a legal flow, as it violates skew symmetry and flow conservation at u and v. f' has one more unit of flow entering u than leaving u, and it has one more unit of flow leaving v than entering v. The idea is to try to reroute this unit of flow so that it goes out of u and into v via some other path. If that is not possible, we must reduce the flow from s to u and from v to t by one unit. Look for an augmenting path from u to v (note: not from s to t).

- If there is such a path, augment the flow along that path.
- If there is no such path, reduce the flow from s to u by augmenting the flow from u to s. That is, find an augmenting path $u \rightsquigarrow s$ and augment the flow along that path. (There definitely is such a path, because there is flow from s to u.) Similarly, reduce the flow from v to t by finding an augmenting path $t \rightsquigarrow v$ and augmenting the flow along that path.

Time: $O(V + E) = O(E)$ if we find the paths with either DFS or BFS.