

CSE 5311 Lab Assignment 2

Due July 23, 2008

Goals:

1. Review of network flows.
2. Understanding of edge connectivity.

Requirements:

1. Write (and test) a C/C++ program to compute the *edge connectivity* (CLRS, problem 26.2-9) of an undirected graph with no more than 500 vertices and 100,000 edges. The first line of the input (read by shell redirection) will be n and m , the number of vertices and edges, respectively. The remaining lines will contain unordered pairs, one pair per line, for the m edges. Vertex numbers will be in the range $0 \dots n - 1$.

Your output must include 1) the edge connectivity value for the graph, 2) a pair of vertices that yields the “minimum max-flow” when those vertices serve as the source and sink of a unit-capacity (directed) flow problem, and 3) the edges across the minimum cut for the flow problem of (2).

Your program must compile and execute on OMEGA. There should be a comment near the beginning of your code that indicates how to compile on OMEGA. Your debugging trace should be disabled in the version you submit.

2. Email your code and report (as attachments) to mzmiah@uta.edu before 10:15 am on July 23. The subject should include your name as recorded by the University.

Getting Started:

1. You may use netflow code from the course webpage or elsewhere. Your code must use adjacency lists, not an adjacency matrix.
2. To achieve the bound on the number of maxflow instances indicated in CLRS, you should use the same vertex as the source for all instances, while using every other vertex as the sink for one of the $n - 1$ instances. In particular, the chosen source vertex should be one of minimum degree to assure that the edge connectivity is found in $O(m^2)$ time.