York University CSE6117 September 22, 2014

Homework Exercise #3
Due: 11:30 a.m., October 1, 2014

1. Consider the convergecast problem in a totally asynchronous message-passing system. There are n nodes (processes) in the system and m edges, forming a connected graph. Assume that processes have unique identifiers, each of which is $O(\log n)$ bits long. One process has the identifier 1. Processes have no initial knowledge of the network topology: each node only knows how may edges are incident with it.

(a) Assume there are no failures. Each process starts with an input value that is either 0 or 1. The process with identifier 1 must eventually perform an *output*(S) action, where S is the sum of all the processes' input values. Give an algorithm to achieve this. Your algorithm should use $O(m)$ messages, where each message can be $O(\log n)$ bits long.

(b) Now, assume processes may experience halting failures, but enough processes remain non-faulty so that the network of non-faulty processes never becomes disconnected. Again, each process starts with an input value that is either 0 or 1. In any execution where the process with identifier 1 does not fail, it must eventually perform an *output*(S) action, and when that action occurs, S must satisfy the following two constraints.

- S is less than or equal to the sum of all processes' inputs.
- S is greater than or equal to the sum of all non-faulty processes' inputs.

Is this problem solvable? Show your answer is correct.