Computational Methodologies: Distributed Computing

bcng57

Question 1

(a)

Code for each node. Initially, *asleep* = `true` for all nodes. *in* is the binary input, n is the total number of nodes

```
upon receiving no message :
  if $asleep$ then
    $asleep$ = false
    send $\langle in, 0 \rangle$ to left

upon receiving $\langle bit, count \rangle$ from right :
  if $count$ == $n$:  // If visited nodes == total number of nodes, end
    result = $bit$
    terminate
  else:
    send $\langle bit$ & $in, count + 1 \rangle$ to left
```

(b)

Code for each node. Initially, *asleep* = `true` for all nodes. *in* is the binary input, n is the total number of nodes.
This algorithm has $\lfloor n/2 \rfloor$ rounds numbered $1, \ldots, \lfloor n/2 \rfloor$.

```
upon receiving no message :
  if $in$ == 0 then
    result = 0
    send $\langle terminate \rangle$ to left and right
    terminate
  else if $asleep$ then
    $asleep$ = false

upon receiving $\langle terminate \rangle$ from left (resp., right):
  result = 0
  send $\langle terminate \rangle$ to right (resp., left)
  terminate

if no messaged received by round $\lfloor n/2 \rfloor$:
  result = 1
  terminate
```