

**Due date:** Tuesday Dec 9.

Start each problem on a new page.

Write clearly or type

Use only one side of the paper (if hardcopy)

Letter size paper (just in case)

## Problem 1

Define a **reliable broadcast channel** to be a channel such that:

- every message sent is eventually delivered to all addresses (including the sender's address)
- all addresses receive messages in the same order, which can differ from the global send order (so if  $v1.tx(m1)$ ,  $v2.tx(m2)$ ,  $v3.tx(m3)$  occur in order,  $m1, m2, m3$  can be received in any order).

Develop a service program for such a channel. A skeleton is provided below.

```
service RelBroadcastChannel(Set ADDR) { // reliable broadcast channel
  ic {ADDR.size ≥ 1}
  ....
  return v; // v[j]: access system at j
  input void v[ADDR j].tx(Seq msg) {...} // nonblocking
  input Seq v[ADDR j].rx() {...}
  atomicity assumption {...}
  progress assumption {...}
} // RelBroadcastChannel
```

**Example evolutions** Here are some acceptable and unacceptable evolutions for a channel of addresses A, B, C, D.

Evolution 1: ACCEPTABLE

A sends a1

B sends b1

C rcvs b1

C rcvs a1

D rcvs b1

Evolution 2: UNACCEPTABLE

A sends a1

B sends b1

C rcvs b1

D rcvs a1

Evolution 3: ACCEPTABLE

A sends a1

A sends a2

B sends b1

B sends b2

C rcvs b1

D rcvs b1

D rcvs a1

D rcvs a2

C rcvs a1

Evolution 4: ACCEPTABLE

A sends a1

A sends a2

B sends b1

B sends b2

C rcvs b2

C rcvs a1

C rcvs b1

D rcvs b2

D rcvs a1

D rcvs b1

Evolution 5: UNACCEPTABLE

A sends a1

A sends a2

B sends b1

B sends b2

C rcvs b2

C rcvs a1

C rcvs b1

D rcvs b1

## Problem 2

Give an **algorithm-level** distributed program that uses the timestamp mechanism to implement the reliable broadcast channel from problem 1. (Hint: Associate an extended timestamp with every user message sent, and deliver user messages in order of extended timestamps.)

Your answer will consist of two programs (in SESF notation or similar), say RbDist and Rb (analogous to the two programs in the timestamp-based distributed lock implementation).

- RbDist starts a fifo channel and an Rb system at each address. (No thinking required here.)
- Rb is an algorithm-level program (i.e., variables and atomic rules).